ENHANCED HYBRID MACHINE LEARNING MODEL FOR DETECTING DENIAL OF SERVICE ATTACKS IN INTERNET OF THINGS NETWORKS

BEATRICE NJERI NGUNYI

A THESIS SUBMITTED TO THE DEPARTMENT OFF COMPUTER SCIENCE AND IT IN THE SCHOOL OF SCHOOL OF COMPUTING AND MATHEMATICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF MASTER OF CYBER SECURITY OF THE CO-OPERATIVE UNIVERSITY OF KENYA.

2025

DECLARATION

Declaration by the candidate

This thesis is my original work and has not been presented for the award of a degree in any other university.

…… ……21/11/2025……

Signature Date

NGUNYI BEATRICE NJERI

MCSC01/6001/2022

Declaration by the supervisors

I/We confirm that the work reported in this proposal/thesis was carried out by the candidate under our supervision and has been submitted with our approval as university supervisors.

…… ……21/11/2025…….

Signature Date

Name of Supervisor, Department, School and University

Dr. David Muriuki

School of Computing and mathematics

The Co-operative University of Kenya

…… …… ……21/11/2025……

Signature Date

Dr. Andrew Omalla

School of Computing and mathematics

South Eastern kenyan University

ACKNOWLEDGEMENT

I thank the almighty God for giving me the strength, patience, and energy to endure the difficult and tedious times I have had during my study. I am heavily indebted to various people who provided material and non-material support for this study to succeed. My appreciation goes to my children for their continuous support and patience during my days of study.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

IOT	Internet of things
DOS	Denial of service attacks
DDOS	Distributed Denial of service attack
ML	Machine learning
RF	Random Forest
IF	Isolation Forest
XGBOOST	Extreme gradient Boosting

OPERATIONAL DEFINITION OF TERMS

DoS Attacks is a cyberattack in which an attacker attempts to make a computer, network, or online service unavailable to legitimate users

Hybrid Machine learning refers to any system or approach that combines two or more different machine-learning methods, models, or techniques to achieve better performance than each method could achieve alone.

Intrusion Detection Systems is a security tool that monitors network traffic or computer systems for suspicious activity, policy violations, or signs of cyberattacks and alerts administrators when such activity is detected

IoT Security refers to the practices, technologies, and strategies used to protect Internet-connected devices and the networks they operate on from cyber threats

Supervised learning is a type of machine learning in which a model is trained using labeled data—data that includes both the input features and the correct output (target).

Unsupervised learning is a type of machine learning in which a model is trained on data that has no labels. The system tries to find hidden patterns, structures, or relationships in the data without being told the correct answers.

Edge Computing is a computing model in which data processing and storage happen close to the source of the data—such as sensors, IoT devices, or local servers—rather than relying entirely on distant cloud data centers.

Scalability refers to a model's or system's ability to handle increasing amounts of data, computational load, or complexity without a significant drop in performance.

ABSTRACT

Connecting devices across homes, healthcare, agriculture, transportation, and businesses through the Internet of Things (IoT) has become a critical part of modern life. Nevertheless, the universal interconnection of IoT devices introduces significant vulnerabilities, making them susceptible to cyber threats such as Denial-of-Service (DoS). These attacks deny legitimate users access to services and cause financial losses due to Service Level Agreement (SLA) violations. Traditional approaches have proven to be insufficient in handling large-scale, diversified, and complex IoT environments. This study developed an enhanced hybrid machine-learning model that integrates Isolation Forest (unsupervised anomaly detection) with Random Forest (supervised classification) to improve the detection of DoS attacks in an IoT network. Benchmark datasets, including NSL-KDD and CICIDS017, were obtained from the Kaggle open-data repository, where the complete datasets were collected using Octoparse web-scraping software. Octoparse enabled the automated extraction of the entire dataset without modification, ensuring that all available records were included in model training and evaluation. Experimental results demonstrated that the hybrid model achieved superior performance – Accuracy (97.8%), a Precision of 98.2%, Recall (96.8%), F1-score (97.5%), and ROC-AUC (98.1%) consistently outperforming standalone models. The ROC curve analysis confirmed the hybrid model's discriminative strength, showing clear separation between normal and attack traffic. In addition, scalability tests revealed that the model scales effectively to large datasets, with projections up to 5 million records. These findings highlight both the consistency and superiority of the hybrid model compared to traditional ML-based IDS solutions. Academically, the research fills a gap by emphasizing hybrid ML approaches specifically tailored for DoS detection in IoT environments and practically delivering a robust, scalable, and real-time security solution for industries and organizations seeking to secure large-scale IoT infrastructures.

CHAPTER ONE: INTRODUCTION

1.1 Background of the Study

Denial of Service (DoS) attacks primarily disrupt system operations by generating numerous unnecessary requests. As a result, users are unable to access or interact with IoT devices, making it challenging to make informed decisions. Moreover, these attacks compel IoT devices to remain continuously active, resulting in a reduction in battery lifespan. A Distributed Denial of Service (DDoS) attack is a cyber-attack involving multiple sources, each with distinct IP addresses, sending the server a massive volume of requests simultaneously. This overloads the system, making it challenging to identify and separate genuine traffic from malicious requests. (Devi & Kumar, 2015) In recent years, a distinctive IoT botnet known as Mirai has launched powerful DDoS attacks, disrupting thousands of IoT devices by interfering with their normal operations (Islam *et al.,* 2022). The Denial of Service (DoS) attack works by preventing users from accessing the server by the attacker through disrupting its normal functionality. Common examples of such attacks include Neptune, Ping of Death, and Mail bomb.

IOT devices make up objects made up of communication sensors, actuator modules that connected through the internet. IOT growth is rapid across the world, forming the smart systems of networks. Intelligent systems are the objects incorporated with smart devices and are used daily and function without the interference by human intervention (Jayasree et al 2020). Cloud computing, big data among other techniques enable the IOT devices to gather data, process and transmit it from one device to another. Thus, transmission of information across the world has been made possible with the use of network protocols. (Merzouk, *et al*., 2020).

Hybrid machine learning models offer a more efficient method for strengthening network security within IoT environments. Through behavior-based analysis, they are capable of identifying irregularities in network traffic that may indicate botnet activity, leading to higher detection accuracy. Additionally, these algorithms incorporate a feedback loop that enables ongoing refinement by retraining on previously flagged false positives (Venkatesan *et al*., 2023).

1.1.1 Intrusion Detection

The most commonly used security mechanisms are Intrusion Detection Systems (IDS) (Arshad *et al*., 2020). Intrusion Detection Systems assume anomalous traffic differs from normal traffic; thus, it is separable and can be detected. Monitoring inbound and outbound traffic, anomaly detection, and notification are the main activities of IDS. IDS is categorized into Network Intrusion Detection Systems (NIDS) or Host Intrusion Detection Systems (HIDS) (Diro & Chilamkurti, 2018; Ajaeiya *et al*., 2017)

1.1.2 DoS attacks

According to Kimmi *et. al*., 2022, the attack that leads to the inability of genuine applications to access network infrastructure by malicious users is identified as a Denial of Service. Also known as an effort that limits access to the network and information to the users. To overwork the resources of the targeted victims, the network builds by generating heavy traffic, ensuring that vital information is not captured, and compromising files and credentials. Data packets traversing to different destinations is made impossible by the heavy traffic thus a three-hand way handshake is important for a network connection to be established. Assigning a reservoir connection is necessary, which is done after establishing connectivity between a client and a server. The recipient has to acknowledge the connectivity showing the process is complete. The attacker brings forth some similar connections and sends photons as required but does not receives notifications from the provider. Consequently, the association does not mature thus bringing the host to a space where all unfinished transactions are stored. Chances of increasing wealth are minimal, unveiling the host to the storage space allocated for all unfinished transactions. The server buffer will overflow with unresolved attachment almost denying TCP connections to take place thus making it impossible for the server to undertake a DOS strike. DDOS attacks are the hardest threats for security to notice and lessen. This is brought about by the conventional networking devices, the diversity of attack strategies, and the programmer's opacity towards the host's site.

1.2 The Statement of the Problem

Cybersecurity in the Internet of Things networks involves the protection of interconnected smart devices and data interchange from unauthorized access, cyberattacks, and vulnerabilities to enhance reliability, confidentiality, and privacy. The Internet of Things (IoT) works in many aspects of human life, including homes, transportation, agriculture, healthcare, and businesses.

However, the growing interconnectedness of these devices has raised concerns regarding the security and trustworthiness of IoT communications. As IoT systems become more complex and extensive, ensuring secure and reliable interactions among devices has become a paramount requirement (Mosenia & Jha, 2017).

IoT systems are prone to a wide range of security attacks, inclusive of the most common Denial of Service attack denies legitimate IoT network users from accessing services causing disruptions of services thus leading to huge financial losses for industries. Intensive techniques are therefore required to enhance security in IoT networks (Verma & Ranga, 2020; Li *et al*., 2021).

Adoption of Intrusion detection Systems is on the rise to detect and reduce intrusions in IoT networks. The application of machine learning (ML) and deep learning (DL) for intrusion detection in IoT is on the rise (Adefemi Alimi *et al*., 2022; Al-Garadi *et al*., 2020). This arose from the inefficiency and insufficiency of the traditional IDSs for the reduction of cyber intrusions. However, the complex data structure has introduced irrelevant features and outliers in datasets, which is a key challenge in Intrusion detection systems. The proposed solution to address these challenges is the hybrid model that combines both supervised for feature selection and unsupervised for data reduction approaches (Ullah & Mahmoud, 2020; Alzubi *et al*., 2022). Hybrid ML models that combine feature selection (supervised learning) with data reduction (unsupervised learning) approaches have been proposed as promising solutions to address the challenge (Ullah & Mahmoud, 2020; Alzubi *et al*., 2022).

Several studies have been carried out on Machine learning and Deep learning techniques for Intrusion detection on IOT networks. Sahu and Mukherjee (2020) implemented Artificial Neural networks on NSL-KDD99 dataset for the detection of IoT attacks. DoShi Apthorpe and Feamster (2018) investigated packet-level ML detection of IoT networks. Soe *et al*. (2020) designed algorithms for selecting optimal features per attack type to improve prediction. Ge et al. (2019) applied deep learning models to IoT environments to improve detection time and threat prediction. Alsamiri and Alsubhi (2019) explored ML-based detection and classification for IoT networks. Khraisat *et al*. (2019) developed hybrid ML models for detecting network and zero-day attacks, while Ullah and Mahmoud (2020) implemented real-time hybrid models to

categorize IoT network attacks. More recently, Hussain *et al* (2022) and Khan *et al* (2023) emphasized scalable ML-based IDS frameworks for IoT-enabled smart cities, highlighting the need for adaptive models against evolving threats.

While many scholars have researched ML algorithms for IoT attack detection, limited emphasis has focused on developing an enhanced hybrid ML model specifically for DoS attack detection in IoT networks. Therefore, this research seeks to fill this gap by addressing the question: How will an enhanced hybrid machine-learning model be developed to improve the detection of DoS attacks in IoT networks?

1.3 Main Objectives

The study primarily aims to develop an enhanced hybrid Machine-learning model for detection of DOS attack in IOT networks.

1.3.1 Specific Objectives

i. To identify relevant features from IOT network traffic that expose them to DoS attacks.
ii. To design the enhanced hybrid machine learning model.
iii. To investigate the scalability of the enhanced hybrid machine learning model.
iv. To evaluate the operational capability of the enhanced hybrid machine learning model.

1.3.2 Research Questions

i. Which features exist in the IOT network traffic that expose them to DoS attacks?
ii. How will the model be designed/developed to detect DOS attacks?
iii. How will the developed model adapt to an expanding IOT network?
iv. How will the operational capability of the enhanced hybrid machine-learning model be evaluated?

1.4 Scope of study

The study focuses on the detection of DoS attacks in IOT networks using enhanced hybrid Machine learning algorithms. According to Ardabili *et al*. (2020), Hybrid methods combine two different single methods into one method with a higher performance capability and flexibility

than a single method. A hybrid method consists of one prediction unit and one optimization unit, predicted to acquire an accurate output.

1.5 Significance of the study

To future academicians and researchers, the study will provide information on the identification of security breaches and estimating the chances of their occurrence in an IOT network. They will also be able to identify the knowledge gained from the study. The developed algorithm can be implemented in practical IOT devices by industries. The study will provide comprehensive literature on IOT and DOS attacks, which can be used by academicians and researchers.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The literature review to be explored focuses on IoT networks, DoS attacks, and machine learning algorithms for DoS attack detection, including related studies, a conceptual framework, and gaps in the current algorithms that the study seeks to address.

2.2 Theoretical Framework

2.2.1 Computational Learning Theory

The computational learning theory (Michael, 1990) can be defined as the mathematical study of efficiency in learning by machines or computational systems. The demand of efficiency is one of the primary characteristics distinguishing computational learning theory from older but still active areas of learning theory encompassing a wide variety of interesting learning environment and formal models, too numerous to detail in any single volume.

Computational Learning Theory's emphasis on computational efficiency aligns with the constraints inherent in IoT environments. Many IoT nodes possess limited processing power, storage capacity, and energy resources, necessitating algorithms that offer low computational overhead without sacrificing detection capability. Hybrid models such as combinations of clustering with classification or deep representation learning with lightweight ensemble methods are chosen because Computational Learning Theory demonstrates their potential to learn effectively under such constraints

2.3 Empirical Literature

2.3.1 Denial of service attacks (DOS)

DDoS/DoS attacks arise when targeted servers are flooded with a massive number of requests, weakening the target server and ultimately incapacitating it, thereby interrupting services to authentic users. The attacker uses various resources to flood the targeted server. The DOS attacks are not specific to IOT networks. Out of the diversified and complicated nature of IOT networks, the network layer is susceptible to DOS attacks. The weak configuration in IOT devices in IOT

applications has brought about the possibility of attackers unleashing DDOS attacks on targeted servers (Kolias *et al*, 2017).

2.3.2 Internet of Things (IOT)

Internet of Things is described as tangible objects connected together equipped with minimal estimation, repository, and information passage competencies, also coated with sensors and actuators, with the support of network connections that support these things to gather, infrequently process, and pass each other data. The things range from smart devices, including smart ovens, smart refrigerators, smart bulbs, smart adapters, to heartbeat detectors, Radio Frequency Identification (RFID) devices, automobiles, sensors in a parking lot, accelerometers and a group of other sensors in, Retail management, etc., which are more refined. (Hussain *et al*., 2020).

IOT covers several areas like manufacturing, transportation, Business management etc. The enormous scope of IoT networks takes novel tasks such as computation, complete data, storage, communication, privacy and security. A very wide areas of IOT has been looked at by several researchers these areas include Communication, architecture, protocols and privacy (Bahga & Madisetti, 2014). More so, for a consumer to be fulfilled with the IOT technology, security and privacy must be looked into (Hussain *et al*., 2020).

2.3.3 Effect of DOS attacks on IOT networks

According to Sinha 2021, DOS attacks have numerous effects on different layers of the network, including jamming. This is a situation where data is discharged at normal intervals by the attacker. When an attack imitates a network node that is legal and passes data continuously, the action is referred to as flooding. The other module searches for novel nodes; it also identifies attackers whose degree of accuracy is high. Secondly, we have Intrusion, which includes interfering with network activity where an intruder produces network traffic in radio frequency form either every day or on a specified basis. Tampering with the node and destroying it is another effect of DOS attacks, where an attacker improves availability to a node and makes it inactive, or accesses its memory purposely to alter data that guarantees the correct process, identified as compromising the node. Connection is terminated once a node is damaged, when two nodes send data packets on identical frequencies collision takes place. For a malicious node to interrupt the legal nodes' packet entrance at the receiver, data is sent simultaneously as

legitimate nodes. Because of endless collisions, exhaustion takes place, which triggers the channel to become congested. The attacker directs a huge number of requests to send. Regular operations are disrupted when a consequence of unapproved usage of the link-layer system takes place. An attacker node exhibiting different characteristics from other network nodes leads to spoofing. It would be challenging an attacker to transmit such an attack in a network where a group of neighboring nodes utilizes an exclusive key to startup or continuous leaping for a huge network.

2.4 Machine learning approaches to DOS attacks

There are various ML approaches, and different classification methods have been recently given by researchers (Mosavi *et al*., 2018; Mosavi & Varkonyi-Koczy, 2017). Dividing the methods in three groups is one way of classifying ML methods. These include single methods, hybrid methods, and ensembles (Cheng & Yu, 2019; Chou & Tran, 2018). According to (Ayodele, 2010) classifications such as Linear Classifiers, Logistic Regression, Naïve Bayes Classifier, Perceptron, Support Vector Machine, Quadratic Classifiers, K-Means Clustering, Boosting, Decision Tree, Random Forest (RF); Neural networks, Bayesian Networks etc are handled by supervised machine learning algorithms Setiono & Leow, (2000) and Kaufmann, (2005) notes that Supervised machine learning are applied on numerous areas. SVMs and neural networks are superior in terms of performance when working with multidimensional features that are continuous. Performance is considered better for discrete /categorical features, logic–based systems. Minimal storage space is required during training for and classification stages for NB, which uses a small dataset to achieve a maximum prediction accuracy. Neural network models and SVMs need huge sample sizes. K-NN's performance clearly shows that it is sensitive to irrelevant features. Its training phase and execution require a large storage space.

The researcher continues to point out that ANNs and SVMs perform better if there is an existence of multicollinearity more so, if input and output features possess a nonlinear relationship. Though rules and algorithms cannot be used as incremental learners, Naive Bayes and the kind can. Naïve byes has no effect in decision-making are not used in computing probabilities this makes it robust to missing values. On the other hand, complete records are required by KNN and neural networks require performing tasks. Lastly, in terms of operational profile Decision Trees and NB possess profiles that differ, in that one is very accurate the other is not. Contrarily

rule classifiers and decision trees possess matching operational profile, same to SVM and ANN. Other algorithms cannot be out performed by single learning algorithm.

Unsupervised Learning (UL) is described as a machine learning method that detects layouts in datasets with unmarked or unsystematic data points. For this approach the artificial intelligence acquires the data input that isn't related to the output. To oversee the model unsupervised learning does not require the existence of a person. (Alloghani *et al*., 2020). By observing data and detecting patterns the system is permitted to learn on its own. In comparison to supervised learning models, unsupervised algorithms are superior at performing difficult tasks than supervised learning algorithms. Though for supervised learning models the output results are more precise as the programmer needs to openly show the system what to examine in the data presented. It is assumed that deep learning has been made possible by artificial neural networks which is founded by Unsupervised learning (Krotov & Hopfield, 2019).

K-means algorithm a clustering method, works by identifying centroids that lessen withincluster sum-of-square which is also known as inertia. According to (Zeng *et al*., 2022) and (Sharafaldin *et al*., 2019), the clustering problem is addressed by K-means, a modest unsupervised learning techniques. It simply classifies data sets using a predefined number of clusters (k clusters) by following specified approach. This method is said to be suitable when the label data isn't known. (Smola & Vishwanathan, 2008)

Variational Auto encoder (VA-Encoder) and neural network-based, applies trodden representation of unprocessed data (Jolliffe & Cadima, 2016).The VA–encoder makes up the encoder, decoder and loose function. The expected output by the VA-encoder is a known method to clarify review in the latent space. The capacity for VA encoder to evade over fitting challenges provides an assurance the latent space has good features with generative process being formed by the VA-encoder (Bock & Weiß, 2019), (Khoei *et al*., 2022)

Reinforcement ML techniques interact with dynamic environments which can also possess extra machines, that have the ability to sense, perform, accept rewards and also learn. Techniques accepts inputs, gives out outputs and takes in rewards. Its goal is to make maximum use of rewards in respect of future and present actions from the other machines (Ghahramani, 2004)

2.5 Hybrid Algorithms

ML classification methods are vary, and researchers have identified different classification methods (Mosavi & Varkonyi-Koczy, 2017). The classification is divided into three groups, namely single classification method, hybrid classification method, and ensembles classification method (Cheng et al., 2019; Chou & Tran, 2018).

The practice of Combining ML models with different models, and/or other soft computing and optimization methods is done for achievement of the threshold of hybrid models and to expand the model in various aspects, Bagging or boosting, which are grouping techniques that use more than one classifier, make up the ensemble methods. For an ML to succeed, it highly depends on the enhancement of ensemble and hybrid techniques (Singh et al., 2018; Jaiswal & Malhotra, 2018).

Hybrid Algorithms are an amalgamation of a minimum of two algorithms to enhance accuracy, efficiency, or robustness of algorithms. Hybridization combines the benefits of more than two or more algorithms. Developing an algorithm that is able to make informed decisions depending on specific data can be effective as long as the data is noisy, incomplete, or inaccurate. Thus, to enhance the correctness of machine learning algorithms that involve multiple decisions, having a weighted voting system is crucial. Additionally, training an algorithm on a dissimilar data set to make a dissimilar decision is important (Atefi *et al*., 2013). According to Ardabili *et al*., (2020) hybrid techniques can sometimes consists of one prediction unit and the other for optimization of the predicted unit to reach an accurate output. Hybrid methods consist of a combination of various single methods forming a method that has a high capability and flexibility in comparison to other single methods.

2.6 Ensemble learning approaches

The two main methods towards more precise and dependable ML methods are ensemble and hybrid ML methods. According to Tien Bui *et al*. and Zhang & Mahadevan, (2019) Ensemble methods of machine learning provides an optimal model for prediction by merging diverse base models. Ensemble techniques deliver an appropriate model that can be used for forecast by use a mass of models into account and also averages these models. The most current and most

prevalent models are bagging and boosting, while the greatest implementation being with random forest and AdaBoost.

2.6.1 Bagging

This is an ensemble technique that performs final classification by voting whereby models are supplied with random subsets of data set. RF is a bagging technique and consists of different subsets of the dataset from several decision trees and produces their mean to increase prediction exactness (Pandey & Mishra, 2023).

2.6.2 Boosting

Boosting is classified as an ensemble learning technique that generates additive models using weak learners. Boosting is founded on the notion that weak learners turn out to be better. AdaBoost classifier produces a strong classifier by consecutively merging several weak models. The wrongly anticipated data of one model is delivered to the model that follows and so on (Pandey & Mishra, 2023)

2.6.2.1 Extreme Gradient Boosting

According to Kumar et al. (2021) Extreme Gradient Boosting is an ML Hybrid approach built on the boosting. It is a part of a boosting algorithm that transforms weak learners to strong ones. By use of this approach and to accomplish higher weight tasks to the previous trees point of misclassification, the tree is enlarged sequentially, and as it replicates it attempts to decrease the misclassification rate. Additionally, to avoid over fitting it more efficiently accommodates diverse forms of sparsity designs in data input, as well as regularization terms. The objective of regularization is to output a model that is less complex in order to lessen bias and probable overfitting.

2.6.2.2 Gradient boosting

To answer classification and regression tasks, an ensemble algorithm known as Gradient Boosting was established. It combines weak learners into a solitary, effective learner. In order to make ultimate model predictions, the GB-DT trees are joined, though they run independently, thus generating independent dependent forecasts. The number of the estimator’s parameters determines the weak learners. Prediction is unified through picking the class label (MTM in MTM dataset and DOS, in DOS data set) in classification tasks like identifying MTM and DOS,

that has the majority polls based on the entire population of trees (Salih 2014; Mirsky *et al*., 2019).

2.6.2.3 Stacking

Stacking generalisation, also known as stacking, is mostly utilised in ensemble models, where we can create a new model that will be trained to combine predictions from two or more previously trained models. Existing models or sub-models' predictions are merged using the new. It is different from bagging and boosting which to acquire final prediction combines outputs from different learners. Its performance is different from bagging and boosting, which brings together results of various learners to achieve final prediction. Stacking is a diverse learning technique which pools varied base learners through training a model. If appropriately planned, it takes advantage of diverse base learners and expected to outperform other single base learners, whether using the highest voting or weighted averaging (Ting & Witten, 1999; Cho *et al*., 2020; Lv *et al*., 2022; Naimi & Balzer, 2018). Advancement of novel ensemble and hybrid methods determines the imminent success of ML(Singh *et al*., 2018; Jaiswal & Malhotra, 2018; Kumari *et al*., 2018).

2.7 Proposed hybrid algorithm

To achieve the desired combination, the researcher tends to work on a better hybrid performance by taking full advantage of the two algorithms, Isolation Forest and Random Forest. The researcher will sequentially develop a hybrid algorithm by using the Isolation Forest to detect anomalies for this case identify suspicious traffic. Then pass the flagged samples to a Random Forest classifier for further validation and classification.

2.7.1 Random Forest (RF)

Bin Sulaiman *et al*. (2022) and Khatri et al. (2020) Random Forest can be described as a combining classifier that utilizes and integrates several decision tree classifiers. The core goal of use of many trees is sufficient training of the trees and hence permitting each tree to contribute as a model. After creating the tree, the common one is utilized to integrate the outcome. Decision trees are implemented in such a way that each decision tree dependent on a unique dataset whose distribution is dependable on the whole tree. This specific method possesses the ability to equate

errors in a group population of unbalanced data sets in an effective manner. It might be put into use for classification issues and regression challenges.

RF is considered to be a collection of weak base learners that works by constructing several groups of decision trees to boost the DTs' efficiency and robustness (Breiman, L. 2001). The algorithm is resilient to noise and overfitting and has been applied to several domains, for instance, heart disease classification (Singh *et al*., 2022) and label ranking (Zhou *et al*., 2018). Furthermore, improving the classifiers performance is done by selecting data nodes randomly. For L number of leaves, the feature space is spitted by the decision tree into L regions displayed as R L depended on classification performance. The most significant parameters for RFC are the total number of trees and leaves. The feature space is maximized for forecasting the decision trees' final output (Sedjelmaci *et al*., 2017) and (Hu *et al*., 2019), whereby the ultimate predicted outcome is dependent on the popular total votes of the trees.

According to Choubisa *et al*. (2022) as compared to other algorithms RF method takes the least training time. The predicted model produces high accuracy and a minimal loss rate. RF performs better when a large data set is missing. Random forest classification takes place as follows, first stage forest generation for the decision tree takes place by use of a particular data set while stage two deals with prediction from the classifier obtained from first stage.

According to Franke *et al*. (2006) the advantages of Random forest include resistant to over fitting, minimum number of parameters to modify, possessing a tree base approach that obviously arranges the structures, the ability to work on huge datasets which consists of various features and also manages existence of various categorical numerical features are some of the advantages that Random forest possess.

2.7.2 Isolation Forest

According to Isolation Forest or I Forest, which constitutes a mixture of I Trees, is a real and common algorithm that deals with many data and considered an effective technique based on an ensemble for anomaly detection. It differs from conventional anomaly detection approaches caused of high accuracy levels, linear complexity time, and the least memory cost. To "Isolate" observations, isolation forest randomly chooses a feature and arbitrarily identifies a division amongst the highest and lowest values of selected features. The structure of the tree denoted by recursive partitioning, where the distance of the route starting from the node of the root to end

node is denoted by tree structure, equivalent to the splits needed to separate a sample. The size of the normality is distance of the path and our decision function, averaged over a forest of such random trees. Random partitioning yields Short paths for abnormalities. Identification of anomalies enabled by a forest of random trees that jointly outputs smaller path lengths for exact samples. The core concept of IF is that it utilizes two major features of anomaly. i.e., anomalies are rare and possess specific characteristics. Data is divided into two parts by IF, as in a binary tree, division is performed to minimize the specific length of a tree or up to when splitting cannot take place. The architecture formed is referred to as an iTree. Samples separated at the beginning of the split or closer to the root of iTree are likely to turn out to be irregular, normal samples are challenging to separate and need much division until they can be isolated (Liu *et al*,2012)

2.8 Related work

Quincozes *et al*. (2023) specified that, much research is being carried out on DOS attacks although detection of DOS attacks still turns out to be big problem. He proposes that the only possibility for identifying DOS attacks is implementation of machine learning techniques. Quincozes et al, thus offered an assessment of two machine learning techniques supervised and unsupervised. The Researcher came up with a conclusion that supervised approaches performed better thus quicker than unsupervised approaches. Alsulaiman and Al-Ahmadi 2021 examined various ML algorithms to study their ability to detect DoS attacks. The model was trained and tested using WSN-DS dataset. Examined algorithms include SVM, NB, J48, RF and NN. The authors thus recommended that the output showed that RF attained a maximum accuracy of 99.72%. Although it was concluded , that J48 attained almost a similar accuracy of 99.66% and outperformed RF by 9% of its processing time. In contrast to RF, J48 was proved to be more effective at handling hindrances to WSN.

Vinayakumar et.al (2019) evaluated closely on various datasets, including KDD Cup 99, WSN-DS, NSL-KDD, CICIDS 2017, Kyoto, and UNSWNB15, on classical machine learning methods. About binary classifications, results indicated that DT, RF, and AB outclassed SVM-rbf, NB, KNN (LR), and logistic regression. Though RF, DT and AB, compared to all others, retained the same performance, signifying the classifiers are extensible and able to notice novel attacks, while other classifiers revealed different ranges on performance. In relation to multiclass classification, DT and RF performed well, and then AB and the other classifiers followed.

Syarif and Gata (2017) used KDD CUP 99 and a mixture of K-NN algorithm and particle swarm optimization (PSO), the result is optimized 2% than usual K-N algorithm. Wang, CAO, and Hng (2020) used KDD Dataset for classification model and worked on IDS and used Convolution Neural Network (CNN). The consequence of this model revealed that IDS-CNN can proficiently detect intrusion for network data stream and also its rate of detection is better than the modern technique. It was discovered that little research has been done on IDS in the IOT networks.

Anthi et al. (2018) introduced an algorithm that detects network probes and modest DoS attacks like (UDP flood, SYN flood) and uses ML classifiers. For it was evaluated that the system to implement exhibited little correctness (high false negative and high false positive and). Hence favorable outcome is not delivered for DOS attack detection.

Altulaihan et al. (2024) developed a hybrid intrusion detection system (HIDS) which detects unknown network traffic from any node for IoT networks. To define the detection of unknown network packets and block unwanted IPs. Datasets of IPs were running contrary to the design before it became an initial DoS threat. To forecast malicious network traffic against IOT networks a several feature selection methods were estimated. Krishnan *et al*. (2021) study includes procedural forward processing, procedural backward processing, and recursive feature elimination (RFE). Dissimilar logistic regression techniques were used for each selection method estimate, including RF, Support Vector Classifier (SVC) and XG Boost all three methods of logistic regression performed well with a high percentage of correctness. It was concluded that the technique can be implemented in supervised learning to foresee attacks on IOT devices.

Verma & Ranga (2020) discussed enforcing anomaly detection against DOS attacks for IOT networks. Evaluation of performance is implemented on extremely randomized trees, RF, classification, gradient-boosting, regression trees, AdaBoost, and multi-layer perceptron all of which are ML classification algorithms. Random research algorithms was applied to define the optimal parameters of classifiers. Area under the ROC curve, false positive rate, sensitivity, specificity and accuracy are factors that determine classifier performance. To bench mark, the classifier UNSW-NB15, CIDDS-001 and NSL-KDD are implemented. To determine substantial variances amongst classifiers implementation of Friedman and Nmenyi post-host tests is done for statistical analysis of performance measure. The classifiers response time was evaluated using RAsberry Pi hardware device. In relation to performance outputs and statistical tests, it

was discovered classification and regression trees and the extreme gradient-boosting classifiers, suit developing anomalybased IDS for IoT network. Khatib *et al*. (2021), studied ML solutions that detect and protect systems from anomalies. To analyze data sampling effect on ML model, several ML classifiers were used. Resampling and balancing the dataset was done, a comparison of different techniques was also performed using a SMOTE method and a study on binary and multiclass cases was also done. The output indicated, when engaging such data type for detection of cyber-attacks on the IoT network, RF, Linear Discriminant Analysis (LDA), DT, foresaw threats where degree of accuracy was high thus the approaches executed better than others. The execution in binary case for Nystrom-SVM, DT and RF techniques was good. The same technique were implemented the first time in IOT network traffic for attack detection. It was discovered that the algorithm detected attacks efficiently when trained with balanced datasets.

Mukherjee & Sharma (2012), researched on threats on smart devices and IoT systems. To detect anomalies in existing data and to integrate real world data and avoid anomalies and attacks in the future a supervised learning model was integrated.350 k datasets were used to forecast anomalies and to compare performance; high performing ML models are used. Two different approaches are used to perform data analysis. At the beginning, the entire dataset was implemented to the ML models, data points with binary values (0 and 1) in the feature "value", were later eliminated on applying the same classification algorithms the accuracy of the results were 99.4% in the case one and 99.99% in case two.

Brun *et al*. (2018) developed real-time detection technique against IOT gateways for network attacks. A group of metrics mined from capturing of packets were used by Deep –learning approach to predict a network attack, based on testing output on packet captures that contain attacks, to detect attacks appropriately Dense Random Neural Network (RNN) is implemented.

Tyagi & Kumar (2021) built an IDS that can precisely and inevitably identify the difference between new and real-time malicious traffic which focused on extracts of new feature sets blending BoT-IoT dataset. An IOT –specific lightweight feature was established by the researcher which comprised of seven light weight features in place of making use of current feature lessening methods like principal component analysis(PCA)that are capable to adjust basic meaning of variables. The research indicated four attack types including, DoS, reconnaissance, information theft and distributed DoS (DDoS) can be detected by the seven fabricated features. It was confirmed that supervised ML algorithms like DTs, LR, SVM,KNN,

RF and multilayer perceptron (MLP) are effective thus can be applied in securing IoT. To certify the systems' performance a diversified performance evaluation metrics are used, which includes receiver-operating characteristics (ROC), accuracy, FScore, recall and precision. Based on output DT and RF classifiers did equally well with an accuracy (99.9%), though RF proved to be superior based on other metrics training and testing.

Yihunie *et al*. (2019), used ML techniques to examine anomaly-based IDS.NSL-KDD dataset was implemented on Five ML classifiers. Existing network traffic is not represented by NSL-KDD and because of lack of publicly available dataset; NSL-KDD has been implemented in the research. According to the results, tests carried out on five classifiers, with and without one-hot encoding, RF outperformed the four classifiers by achieving almost a perfect result. The RF model recall was displayed a high performance, demonstrating that it achieved the least number of false negatives.

A deeper and broader experiment was conducted by Meng (2011) to compare the performance with the goal of signifying the practice and exploring the challenges of using NNs, SVM, and DT for detection of network anomalies. Through a discussion and an analysis carried out on the impact of feature selection, the output showed that ML methods have the capability of detecting anomalies if appropriately trained, but performance differs dependent on the algorithm. In real life scenarios implementation of ML schemes is delayed by differences in alarm rates which are false, fitness, and data training. To improve the ML approaches performance, it ought to be implemented in a suitable method. According to Kostas (2018), ML was used to develop an anomaly detection network. Due to its update features and possessing numerous attacks CICIDS2017 dataset was used, and for feature selection the RF regression algorithm was implemented. Out of the seven algorithms used in the application set up all showed high performance. At the application step ML algorithms were applied, all displayed the performance was high. The success rates of the algorithms used are as follows; MLP (83%), (QDA) (86%), KNN (97%) Naive Bayes (86%), ID3 (95%), AdaBoost (94%), and Quadratic Discriminant Analysis RF (94%),

Hasan *et al*. (2019) various ML models were examined to detect threats and abnormalities on IoT networks. SVM, RF, LR, DT and artificial neural networks (ANN) which are ML algorithms were used. Recall, Accuracy, F1score, and precision were applied for metrics performance

evaluation comparison. DT, RF, and ANN scored 99.4% in accuracy. Although RF, DT and ANN have shown to possess similar accuracy, using other metrics RF seem to perform better.

Ramadan & Yadav (2020) developed a hybrid IDS system that predicts attacks on IoT network. The two stages of the system were prepossessing and classification's-KDD dataset was used and data processing executed by encoding, scaling, and eliminating noise. Enhanced Shuffled Frog Leaping (ESFL) is used to extract relevant features

Shurman *et al*. (2019), discussed hybrid-based intrusion detection system (IDS) to detect threats across all network nodes for IoT environments was developed. To determine detection of eccentric network packets and stop unwanted IPs prior to turning out to be a DOS threat, IP datasets on it.

Soe *et al*. (2020), developed an algorithm that boosts prediction functionality through specified feature selection for each kind of attack in an IOT network. ML models were used to assess the feature selection algorithm that could easily detect the threats. However, for each attack type the developed algorithm chose static features, which could be seamlessly be avoided if exposed to the attacks.

The authors El Mrabet *et al*. (2019) did a comparison to check the functionality of known supervised ML models namely Naive Bayes, Support Vector Machine, Decision Tree, and Random Forest for smart grid intrusion detection. On comparison of the performance Random Forest classifier out performed other known techniques. Anthi *et al*. (2019) did a comparison on the efficiency of Decision Tree, Simple Logistic Regression, Naïve Bayes, Multi-layer perceptron, Support Vector Machine, Random Forest, and Zero Rule. The outcome proved in detecting intrusions the other models are outperformed by Decision Tree classifiers.

Thapa *et al*. (2020) performed a comparison on Neural Networks and various Decision Trees for network intrusion detection. For network intrusion detection, Classification and Regression Tree classifier outperformed other models. Another study by Roy & Shin, (2019) evaluated numerous supervised techniques, inclusive of Random Forest, Naïve Bayes, Support Vector Machine, and Extreme Boosting, and their capacity to detect intrusions on smart grid. It was discovered Random Forest and Extreme Boosting models outperformed other models.

Arora *et al*. (2021) examined several supervised models like, Support Vector Machine, Decision Tree, Artificial Neural Networks, K-Nearest Neighbors, Naive Bayes, and Random Forest to detect cyber-attacks. The findings revealed that, in relation to other models, Random forest

demonstrated better results, specifically in the rate of detection, false alarm rate, accuracy, and true positive.

Table 1: Summary of approaches to IOT attacks

Reference	Method	Attack	Pros
Quincozes *et al.* 2023	Supervised and unsupervised machine learning algorithms	DoS attacks	Supervised algorithms indicated a better performance and are efficient than unsupervised approaches
Alsulaiman and Al-Ahmadi ,2021	RF,NB, j48, SVM, and NN	DOS attacks	Researchers recommended RF, since it had the highest accurate result of 99.72%.
Vinayakumar *et al* 2019	Closely linked classical machine learning methods on diverse datasets,	Attacks and anomalies.	In relation to binary classification, DT, RF, and AB

	including NSL KDD,UNSWNB15 ,CICIDS 2017, WSN DS, Kyoto and KDI Cup 99		outclassed classifiers like, , KNN, logistic regression(LR), SVM-rbf, and NB. Over all datasets RF,DT and AB retained similar performance, signifying that the stated classifiers are extensible and able to notice novel attacks
Kostas, 2018	ML methods	Anomaly detection network	Success rates are as follows: Quadratic Discriminant Analysis (QDA) (86%),ID3 (95%),AdaBoost (94%),Naive Bayes (86%), RF (94%), KNN (97%)and MLP (83%),

(Meng, 2011)	Exploring the challenges of using NNs, SVM, and DT	Network anomalies	MLmethods have a capability of detecting anomalies if well trained, performance differs dependent on the algorithm. Delays in extreme application of NL schemes in real life scenarios are brought about by differences in fitness, negative alarm rates and data training
(Yihunie *et al*., 2019), examined using Anthi *et al*.,2018	NSL-KDD dataset was applied to ML classifiers Introduced a model that makes use of ML classifiers for network probe detection and uncomplicated DoS attacks.	anomaly-based IDS	System does not deliver favorable output for DOS detection. It exhibited through intense false positive and intense false negative.
Syarif and Gata (2017) Khatib *et al*., 2021	Used KDD CUP 99 and a mixture of KNN algorithm with		Result is optimized 2% than usual K-N algorithm

	particle swarm optimization (PSO),		
Wang,CAO, and Hng (2020)	Used KDD Dataset for classification model and worked on IDS using Convolutional Neural Network (CNN).	Network intrusion detection	IDS-CNN can proficiently detect intrusion for network data stream, also its rate of detection is better than the modern technique
Altulaihan *et al*., 2024	IP datasets ran against the design	Suspicious network traffic	
Krishnan *et al*., 2021	Feature selection engaged sequential backward and forward processing and recursive feature elimination	Attacks on IOT devices.	The following logistic regression techniques were implemented, including XG Boost, RF and Support Vector Classifier (SVC)
Verma & Ranga, 2020	Evaluation of ML classification algorithms takes place including	DOS attacks using anomaly –based detection	Anomaly based IDS tailored for IOT environment classification regression trees and extreme gradient boosting classifiers are optimal

	gradient-boosted machines,extremely randomized trees, RF, multi-layer perceptron, AdaBoost, classification and regression trees		ML classifiers outperformed others since they can foresee attacks, with the accuracy degree being high.
Khatib *et al*., 2021	Implementation was carried out using Random Forest (RF) , Linear Discriminant Analysis (LDA), and Decision Tree (DT)	abnormal states	
Mukherjee & Sharma, 2012	A supervised learning model was integrated	Attacks in smart devices and IoT systems.	Implementation of Classification algorithms took place,and outputs displayed are first case, 99.4% and second case, 99.99% accuracy.
Brun *et al*., 2018	Deep learning approach		Attacks were detected using Dense Random NeuralNetwork (RNN)

Tyagi & Kumar, 2021	Lightweight feature set lightweight feature set.	IOT attacks	Capability and suitability of supervised ML algorithms inclusive of KNN,SVM, DTs, LR,multilayer perceptron(MLP), and RF RF performs better.
Hasan *et al*., 2019	LR, artificial neural networks (ANN), SVM, DT, RF, ML algorithms are applied.	IoT Anomalies	The accuracy achieved by DT, ANN, and RF is 99.4%.
Ramadan & Yadav, 2020	Implementation of NSL-KDD dataset was done and data processing was executed by encoding, scaling, and eliminating noise.	IoT network attacks	

2.9 Research gap

Although many machine learning and deep learning models—such as CNNs, LSTMs, and XGBoost—have been used to improve IoT security, most studies still rely on just one type of learning approach, either fully supervised or fully unsupervised. Models like CNNs and LSTMs can analyse complex traffic patterns well, but they require far more computing power than most IoT devices can provide. On the other hand, models such as XGBoost may achieve high accuracy but often struggle when the data contains noise or unnecessary features, which is common in real network traffic.

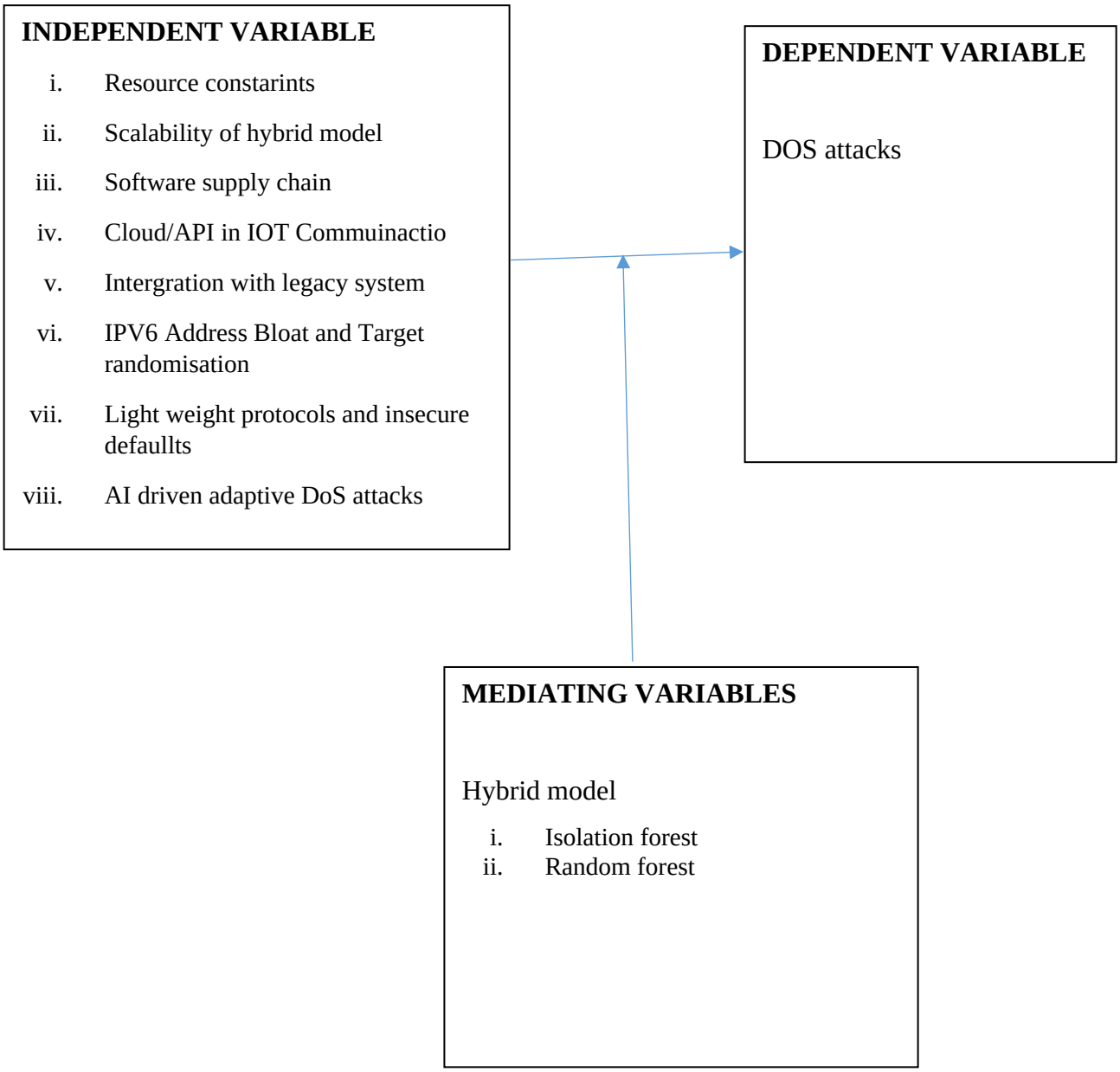
What is missing from current research is a stronger focus on **hybrid models** that combine the strengths of both unsupervised anomaly detection and supervised classification—especially for detecting DoS attacks in IoT networks. The few hybrid attempts that exist rarely examine how well these models scale to large datasets, how effectively they manage feature selection, or whether they can be deployed in the diverse and resource-limited environments typical of IoT systems. Moreover, many studies still do not test their models on widely used benchmark datasets like NSL-KDD and CICIDS2017, leaving questions about real-world reliability and adaptability.

This gap shows the need for a more practical and scalable hybrid approach that can handle real IoT conditions while maintaining strong detection accuracy.

.

2.10 Conceptual framework

Figure 0-1 Conceptual framework



CHAPTER THREE: RESEARCH METHODOLOGY

3.1 Introduction

The focus on this Chapter is Research paradigms and Research design, suited for this research. Hybrid methodology approach was used to develop the proposed model.

3.2 Research Paradigm Positivism

A research paradigm represents a set of shared beliefs and assumptions that guide how problems are understood, how knowledge is generated, and how research is conducted (Creswell, 2003). The research philosophy adopted in this study was the positivist paradigm, also referred to as the scientific or empirical method (Rahi, 2017). This paradigm was considered appropriate because it is based on the assumption that true knowledge can be attained through systematic observation, experimentation, and objective measurement.

3.3 Research Design

A research design provides a blueprint for collecting and analyzing data in order to address research questions effectively. This study employed an experimental research design, which is widely regarded as a scientific methodology that allows researchers to manipulate variables and draw causal inferences. This design enabled the analytical testing of hypotheses regarding the influence of machine learning algorithms on intrusion detection performance. The main stages of the design included data collection, preprocessing (feature selection and discretization), model training, and performance evaluation.

3.4 Target Population

In this study, the target population included every entry found in the dataset collected from Kaggle website. Kaggle is an online community where people can access and share datasets, especially for data-science and machine-learning work. It provides a large library of publicly available data contributed by users, researchers, and organizations. The entire dataset was used so that all patterns, behaviors, and variations could be fully represented. This approach helps ensure that the analysis reflects the complete picture of the data and avoids potential bias that

might come from selecting only a subset. All steps in the study from preparing the data to building and evaluating the models, are therefore based on the full set of information provided on Kaggle.

3.5 Dataset collection

The datasets used in this study were obtained through web scraping from the Kaggle repository. The data consisted of network traffic records related to DoS attacks. These datasets were selected since they are widely used in intrusion detection research and provide labeled attack and normal traffic instances suitable for machine learning evaluation. Web scraping, which is often not available through conventional data sources, was selected though it enables the automated collection of large-scale data in real time, a feature that is not available through conventional data sources (Sirisuriya, 2023). Octoparse software was used for extraction. The software had a visual interface for building web scrapers, thus no need for coding expertise. It allowed the acquisition of data from various websites, and the results exported were in formats compatible with machine learning algorithms (Amarikwa, 2024).

3.6 Ethical Considerations

This study was conducted with strict adherence to national and institutional ethical guidelines governing research, data handling, and cybersecurity studies. The following ethical principles guided the research process.

3.6.1 Regulatory Compliance and NACOSTI Approval

Prior to beginning data collection, the researcher obtained a research license from the **National Commission for Science, Technology and Innovation (NACOSTI)**. This approval ensured that the study complied with Kenyan legal and ethical standards governing scientific research, including requirements for responsible data use, secure storage, and minimization of risk to individuals, organizations, and digital infrastructure. All research activities were conducted within the scope authorized by NACOSTI.

3.6.2 Ethical Sourcing of Data

The dataset used in this study was obtained from the **Kaggle** open-source data platform, which provides datasets for research and educational purposes under clearly defined usage rights. Only datasets that were publicly available, legally accessible, and explicitly permitted for academic use were included. This ensured that the research did not involve unauthorized access to private or proprietary datasets or systems.

3.6.3 Data Acquisition Using Octoparse

The study utilized **Octoparse software** to download the selected Kaggle dataset. Octoparse was used strictly within Kaggle's permitted usage policies and did not involve scraping of restricted, sensitive, or user-specific data. The tool only facilitated access to publicly available data and did not involve interaction with live IoT systems or real user networks. This approach preserved both legal and ethical integrity in the data acquisition process.

3.6.4 Privacy, Confidentiality, and Anonymization

The dataset obtained from Kaggle did not contain any **personally identifiable information (PII)** or sensitive user-specific details. All entries represented anonymized IoT network traffic intended for cybersecurity research. Throughout the study, no attempt was made to identify individuals, trace data to specific users or devices, or reconstruct any sensitive information. The data was securely stored and used solely for academic and research purposes.

3.6.5 Informed Consent Considerations

Since the study relied entirely on secondary, anonymized, and publicly available data, direct informed consent from individuals was **not required**. Kaggle's dataset usage terms serve as a framework for secondary use, ensuring that the data provided is suitable for research without infringing on privacy rights or ethical norms relating to human subjects.

3.6.6 Minimization of Harm and Dual-Use Concerns

Cybersecurity research naturally carries **dual-use potential**, meaning insights could theoretically be used for either defensive or malicious purposes. To minimize this risk, the study focused exclusively on designing a defensive approach—a hybrid machine learning model to detect and mitigate DoS attacks in IoT networks. No vulnerabilities, attack scripts, or sensitive configuration

details that could facilitate malicious cyber activity are disclosed in the research. The intention and execution of the study remain strictly aligned with improving digital security.

3.6.7 Integrity and Security of Research Artifacts

All datasets, trained models, scripts, and system configurations were stored securely in controlled digital environments. Access was restricted to authorized use only. Proper measures—such as encryption, password protection, and secure version control—were applied to prevent tampering, unauthorized access, or misuse of the research materials.

3.6.8 Transparency, Fairness, and Model Accountability

The development and evaluation of the hybrid machine learning model were conducted transparently, with full documentation of algorithms, processes, and limitations. Efforts were made to minimize **false positives** that could unnecessarily block legitimate IoT device activity, as well as **false negatives** that could allow attacks to pass undetected. The study emphasizes that intrusion detection models must remain explainable, auditable, and ethically deployable in real-world IoT environments.

3.7 Pre-processing dataset

Preprocessing of data collected was important to give room for network growth and handling large volumes of data. Preprocessing consisted of cleaning the datasets by extracting unimportant attributes, processing missing values, removing duplicate records, and handling outliers. Conversion of data into a format compatible with machine learning algorithms took place. Preprocessing contributed to the model training process being accurate, efficient, and not having a negative effect to the datasets through noise or inconsistencies (Verdonck *et al*., 2024).

3.8 Discretization

To transform continuous numerical attributes into categorical (nominal) values, discretization was applied. This process was seen to be important since some classification techniques proved to perform better on nominal features than on continuous variables. Discretization simplifies data

representation and makes it easier for algorithms to identify meaningful patterns, hence enhancing decision tree accuracy.

3.9 Feature Selection

To improve model performance, avoid overfitting, and reduce computational costs, Feature selection was carried out. Real-time IoT datasets often contain irrelevant, redundant, and noisy features, to minimize storage requirements and ensure high accurate levels feature selection was applied (Li *et al*., 2018).

This study adopted a hybrid feature selection approach that combined filter and wrapper methods. The filter methods assess the importance of the features. Relevance was achieved by calculating a ranking criterion and identifying low-scoring features that fall below a specified threshold. The common filter-based feature selection methods include information gain, stepwise regression, and principal component analysis (PCA) methods. The advantage of filter methods is that they are computationally efficient and independent of the classification/clustering algorithms.(Chen *et al*., 2020)

Wrapper methods use a predictor (based on some supervised learning algorithm) as a black box, and the predictor's performance is used as an objective function to evaluate the representative of the feature subset. During the search procedure, various feature subsets are generated and evaluated. Training and testing the predictor determines the evaluation of a specific feature subset. The interaction between feature subset searches and predictor selection, and the ability to take into account feature dependencies, are some of the advantages of wrapper methods. Wrapper methods are usually computationally expensive since training and testing the predictor require a certain computational cost. (Chen *et al*., 2020)

3.10 Prediction and detection

A hold-out validation method was used to divide the datasets into training and testing sets. For training Eighty percent (80%) of the data was used, while the remaining 20% was used for testing the performance of the trained model.

The prediction and detection process, which involved a two-stage hybrid model, was used. To compute the anomaly scores, the Isolation Forest algorithm was trained on the input data. Instances identified as outliers were filtered out, outputting a cleaned and enhanced dataset. To train the random forest classifier the refined dataset was used. Integrating these two algorithms enabled more reliable predictions to be made by the Random Forest by learning from data that had already been optimized through anomaly detection.

3.11 Performance Evaluation

Four classification metrics, including correctness, precision, recall, and F1-score, were applied to review the performance of machine learning. Performance was computed using True Positive, False Positive, True Negative, and False Negative.

Accurate prediction of positive and negative instances is known as accuracy. Accuracy will be calculated as shown below;

$$Accuracy = \frac{(TP + TN)}{(TP+TN+FP+FN)}$$

Precision computes the probability of an accurate positive prediction.

$$Precision = \frac{TP}{(TP+FP)}$$

The number of correct classifications is measured by recall and penalized by the number of missed entries.

$$Recall = \frac{TP}{(TP+F\mathrm{N})}$$

F1-measure, which serves as a derived effectiveness measurement was calculated by Harmonic mean of precision and recall.

$$Measure = 2* \frac{(Precision*Recall)}{(Precision+Recall)}$$

The area under the Receiver Operating Characteristic curve (AUC-ROC) was calculated to find the model's discriminative ability. It checked its performance across various classification limits. The AUC values range is 0.5 to 1.0, and better performances are represented by larger AUCs.

3.12 Proposed framework

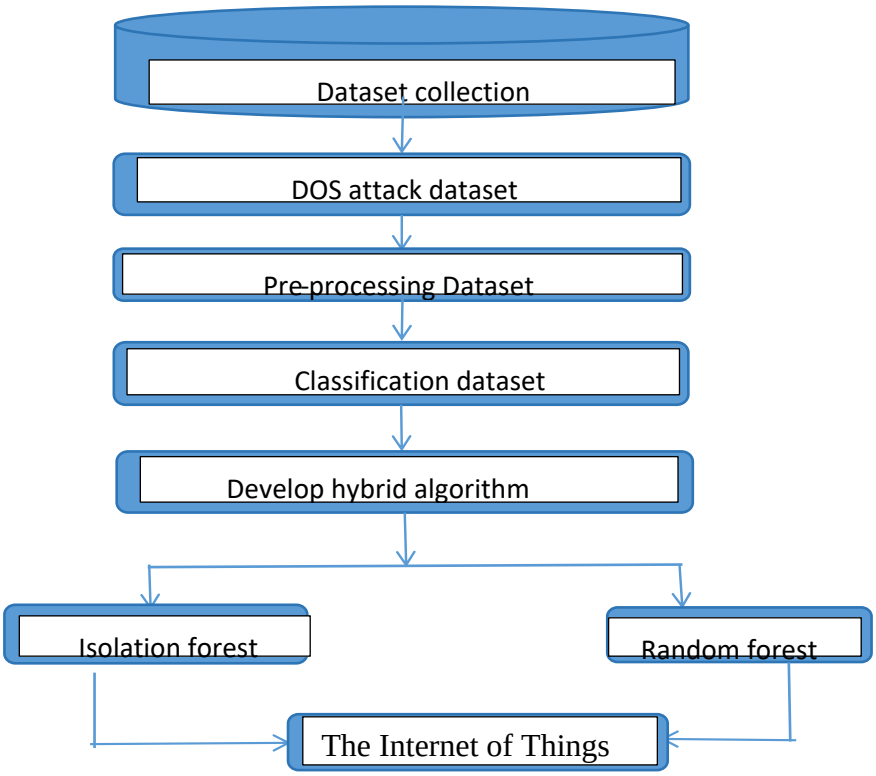


Figure 0-1

The diagram illustrates the proposed framework, which consists of five stages. It shows the interaction between dataset collections, DoS attack dataset, data preprocessing, and classification of dataset, hybrid model development and implementation on the IoT environment. This framework is important because it demonstrates how Isolation Forest and Random Forest complement each other in identifying DoS attack patterns, ensuring both anomaly and signature-based detection. The process begins with data collection from IoT sensors. The data will be normalized and fed into Isolation Forest, which detects anomalies. The output will be combined with Random Forest classification results to improve accuracy and reduce false alarms. By integrating the two algorithms, the model achieves improved detection of unknown DoS patterns, which directly addresses Objective 3—to design the enhanced hybrid machine learning model.

CHAPTER FOUR: DATA ANALYSIS, PRESENTATION AND INTERPRETATION

4.0 Introduction

This chapter presents the experimental analysis, architectural design, and empirical evaluation of the hybrid intrusion detection model developed to counter Denial of Service (DoS) attacks in Internet of Things (IoT) environments. The work directly addresses the study's first two objectives: (i) identifying traffic characteristics and feature patterns that expose IoT systems to DoS exploitation, and (ii) formulating and testing a dual-stage framework that integrates an unsupervised anomaly detector with a supervised classifier to improve detection reliability. The chapter is organized to progressively build from problem context to technical solution. Section 4.1 analyzes modern IoT features and vulnerabilities, illustrating how operational constraints such as device heterogeneity, lightweight protocols, and IPv6-related complexities create opportunities for DoS adversaries. Section 4.2 presents the design of the hybrid detection architecture, highlighting the complementary roles of Isolation Forest (IF) in anomaly detection and Random Forest (RF) in traffic classification, and detailing the workflow of dataset preparation, training, and testing. Section 4.3 evaluates the models through standard performance metrics, including Accuracy, Precision, Recall, F1-score, and ROC-AUC, supported by visual outputs such as confusion matrices, ROC curves, and feature importance rankings. The results are then compared against baseline models to demonstrate the superiority of the hybrid approach. The chapter concludes with a synthesis of findings that validate the model's effectiveness for realtime IoT security applications.

4.1 Characteristics and Vulnerability of IoT Networks

IoT networks are composed of a vast number of interconnected devices that operate autonomously to collect, transmit, and process data. New vulnerabilities have come up as a result of changing architectural, operational, and protocol level characteristics in recent applications, which have led to benign vulnerabilities that have higher effects of Denial of Service attacks. Presentation of the IOT environment and its constraints, and how the hybrid model responds to the constraints, is discussed in this section.

4.1.1 Resource Constraints

Resource constraint is one of the limitations of IOT devices, including minimal processing power, memory capacity and battery life. The importance of supporting long-term, autonomous operations in remote or distributed settings and cost-efficiency are the cause of the choices. Deployment of traditional security protocols for instance real –time encryption, intrusion prevention systems and deep packet inspection hinder the stated limitations (Mourya & Prasad, 2023).

The devices ability to defend itself against DoS attacks is affected directly by the resource scarcity. Through flooding the devices with connection requests, malformed packets or redundant data in large volumes the attackers are able to exploit the device's incapacity to manage high –throughput or malformed traffic. This leads to overloading of the CPU, exhausting the available memory, lowering the battery's capacity, leading to the system not functioning. This is also applicable to applications that run on real-time mode like healthcare IOT or autonomous vehicles, where latency and reliability matter (Rauf *et al*., 2022).

According to studies, vulnerability in constrained devices is because of the absence of power- efficient security mechanisms (Zhou et al., 2023). Risk is also amplified by consuming memory and increasing attack vectors by the presence of background services in majority of the devices (Arora *et al*., 2021).

Future research should focus on developing lightweight, energy-efficient anomaly detection algorithms centralized for low processing power and memory usage to mitigate resource constraints in IOT devices. In order to reduce computational overhead while maintaining detection accuracy techniques like model compression, quantization and edge–based pre-processing should be applied. For remote or battery –powered nodes adoptive sampling strategies and event data collection have the capability of extending the battery life specifically for remote or battery-powered nodes. Low-powered communication protocols like LoRaWAN or BLE should be integrated to support sustainable, long-term IOT operations.

4.1.2 Device Heterogeneity

Extreme heterogeneity, comprising of a large array of devices differentiated by the hardware architecture, firmware versions, operating systems and communication protocols are characteristics of the IOT landscape. This has brought about challenges of enforcement of uniform security despite the fact it promotes flexibility and cost optimization. Different vendors have different devices that may not support similar encryption algorithms, patching methods, or authentication mechanisms. The implementations of a standardized security framework across networks is hampered by this inconsistency (Verma & Ranga, 2020).

The attack increased by heterogeneity through introducing weak links into the network. Attackers using them as entry points for broader attacks target outdated or poorly configured devices. Older devices lacking security patches coexist with newer models exacerbates the challenge. Malformed packets or man –in –the –middle attacks can be delivered by exploiting protocol mismatches. Alsamiri & Alsubhi (2019) argue that, the entire IOT environment is easily compromised by attackers through systematically targeting the least secure node and pivot.

As shown from recent findings inconsistences in authentication and update mechanisms correlates with device heterogeneity strongly (Krishnan *et al.*, 2021). Uniform enforcement of security policies further complicated by integration with legacy and proprietary systems.

Future research should explore the integration of standardized, vendor-agnostic security protocols like lightweight, cross-platform authentication and encryption frameworks to address device heterogeneity in modern IOT networks. Adopting open, interoperable communication standards like MQQT-SN or CoAP with DTLS that can reduce protocol mismatches. The proposed model should be incorporated with federated learning approaches, enabling collaborative model training across different devices avoiding sharing of raw data thus improving detection accuracy and preserving privacy. Weak links in heterogeneous environments should be minimal by implementing mandatory baseline security requirements for IoT manufacturers. The policy makers and industry bodies should work on these.

4.1.3 Lightweight Protocols and Insecure Defaults

Constrained light weight communication protocols such as MQTT, CoAP, and XMPP, basically designed for low latency and reduced power consumption are mainly used by IOT devices. These lightweight protocols lack inherent security features like end –to end encryption, mutual authentication and message integrity. Lack of such security features attackers take advantage by exploiting protocol level flaws through message spoofing , flooding, or session hijacking resulting to exhaustion of resources hence denial of service(Tyagi & Kumar, 2021). Moreso devices shipped with insecure default configurations like unchanged factory, passwords or open ports rarely modified after installation.

The attack surface is widened by the combination of weak protocols and default settings. Through flooding messages and overwhelming broker queues attackers can exploit MQTT is publish –subscribe mechanisms while support for multicast over UDP enables amplification attacks by te support of CoAP's. For brute-force login or botnet recruitment defaults settings are easily harvested Zhang et al. (2020) demonstrates how fertile grounds for large scale distributed denial – of –service attacks, inclusive of Mirai botnet rise as a result of misconfigured implementation.

The improperly secured protocols stacks are a consistent entry point for low attack, Khatib et al. (2021) Such vulnerabilities turn out to be continuous due to manufacturers prioritizing costs and making it easy for implementation over robust security Pandey & Mishra (2023).

The most exploited entry points in IoT–based Denial of Service (DoS) are lightweight protocols and insecure defaults; hence, addressing lightweight protocols and studying insecure defaults is important. Providing a clear threat model for a detection framework to target led to researchers focusing on detecting and mitigating of such attacks in similar, resource-constrained environments, understanding the weaknesses of MQTT, CoAP, and protocols in the same category. This is achieved through a hybrid approach in conjunction with an Isolation Forest for anomaly detection and a Random Forest for classification of patterns.

Alignment of challenges enabled by identifying protocol misuse at an early stage, including Brute-Force attempts and configuration-based attacks, before the escalation

into large–scale service disruption. This ensures that the proposed model is not only theoretically sound but also practically relevant to real-world IoT security scenarios, where insecure defaults and lightweight protocols remain prevalent due to manufacturer trade-offs between cost, performance, and security.

4.1.4 AI-Driven Adaptive DoS Attacks

The emergence of Artificial Intelligence in cyber security has created a two side. While defenders have created more sophisticated detection techniques, attackers are also working on Artificial intelligence for more advanced patterns. To evade detection systems, AI-driven DoS attacks use reinforcement learning and adversarial modeling to adapt packet payloads, timing intervals, and source /destination distribution. Such attacks slowly learn to exploit weak points in IDS logic by being trained to respond to intrusion detection signals. This has rendered static or rule-based DoS defenses ineffective by presenting a new wave of polymorphic and continuously adapting threats (Zhou et al, 2023).

In comparison to traditional DoS vectors that depend on fixed traffic patterns, dynamic DoS attacks create low-and slow or random pattern variations that are difficult to differentiate from legitimate traffic. This leads to high false positives in conventional detection systems and overburdening of anomaly detection frameworks that are dependent on feature assumptions. As irregular traffic is generated due to the event-driven nature of IOT devices, it becomes challenging to differentiate intelligent DoS from environments like IoT in healthcare or automation of industries, where high risks may be experienced due to system failure (Sahu & Mukherjee, 2020).

To combat Artificial intelligence threats, several researchers recommended the use of retraining–capable models that adapt to new traffic to mitigate Artificial Intelligence threats (Yihunie *et al*., 2019). Research has shown that Random Forest can retain high accuracy when combined with an anomaly-detection component for pre-filtering (Ramadan & Yadav, 2020).

A flexible hybrid model that solves AI-enhanced DoS risks through a two-way approach was generated by this study. An unsupervised anomaly detection approach-Isolation

forest serves as the leading detection method carried out by isolating data points that deviates greatly from normal traffic. In the area of DoS detection, it recognizes patterns like sudden spikes in request rates, unknown payload sizes or redundant connections attempts all of which are different from normal traffic. It operates by selection of random features and partitioning data, with anomalies necessitating fewer split to isolate. The detection of both known and unseen DoS variants is enabled by the design without relying on labelled attack data, making it highly effective against emerging and AI-dominated threats. Once suspicious patterns are flagged, the Random Forest completes the process by offering high-accuracy classification. To ensure adaptability, the architecture is inclusive of periodic refraining with up-to-date traffic datasets to ensure adaptability, hence allowing the model to keep track of evolving trends and avoid concept drift, hence the ability to maintain robust performance under sophisticated, AI–driven DoS campaigns.

4.1.5 IPv6 Address Bloat and Target Randomization

The scalability of unknown addresses, which supports massive device implementation, is enabled by the adoption of IPv6 networks. This has brought about new vulnerabilities known as Denial of Service attacks (DOS). This has made traditional network defenses like IP blacklisting and static filtering ineffective. Attackers, hence evading systems dependent on known IP signatures or rate lining (Alam *et al*., 2022), generate spoofed packets from various IPv6 addresses.

In situations where packet processing is minimal the threat is strengthened when these randomized attacks target devices constrained in mesh or low-power wide-area networks (LPWANs). Security policies that often underutilize IPv6 extension headers can be modified to break down packets or overload device routing logic. Latency and jitter, which are crucial to real time IOT systems such as industrial automation and connected health applications, are increased by the address-bloat strategies that also obscure attack origins (Altulaihan *et al*., 2024; Tyagi & Kumar, 2021).

According to Verma & Ranga 2020, he argues that due to the protocol's flexibility and decentralized routing logic,

IPv6-enabled attacks are difficult to mitigate. More so, header manipulation or entropy variance isn't restricted by default configurations in many network stacks (Pandey & Mishra, 2023).

Unique security concerns for IOT networks and scalability are offered by IPV6 due to its vast address space and support for stateless address auto configuration. Traditional IP-based blacklisting or filtering methods have been made ineffective due to the sheer number of potential spoofed addresses. This arises from attackers exploiting the expanded address range to launch random–source DoS or Distributed DoS (DDoS) attacks. IPV6 offers various benefits for scalability, at the same time introducing unique security concerns for IOT networks. This is as a result of its vast address space and support for stateless address auto configuration.

To increase processing overhead on target devices, IPv6-specific extension headers and neighbor discovery mechanisms can be abused. The IPv6 challenge is addressed in this challenge by implementing a behavior-based detection strategy through a hybrid model. The Isolation forest recognizes anomalies in packet density, request intervals, and entropy variations, which cannot be detected regardless of the attacker's source address. The random forest classifies the suspicious flagged traffic using higher-level features like protocol interactions, destination access patterns, and session duration. The model remains scalable and efficient across large spaces by mitigating IPv6-specific random source DoS attacks through focusing on traffic behavior rather than static IP attributes. The hybrid model does not rely on IP origin but checks on connection density, timing patterns, and packet entropy. This study minimizes these challenges through shifting detection to behavior-based anomaly scoring instead of using static IP analysis.

4.1.6 Integration with Legacy Systems

A complex set of security challenges arises from the integration of IOT devices with legacy systems, such as SCADA, programmable logic controllers (PLC), and industrial control systems. These systems lack basic security features like authentication protocols, encrypted communication, or firmware update capabilities, though they predate modern

security frameworks. The need to preserve existing infrastructure and reduce upgrade costs is what drives them towards inclusion in IOT environments, but this comes as a result of resilience.

This leads to a situation becomes difficult when the legacy systems are networked with IOT devices. Insecure bridges can arise because of the heterogeneous integration where attackers can propagate attacks across the system through exploitation of the weakest link. In sectors where rel time operation is essential such attacks are dangerous. Legacy devices lack cryptographic protection complicating tracing or prevention of DoS attacks initiated through them (Yihunie *et al*., 2019). Critical gaps for malicious exploitation are left out by interoperability issues between industrial protocols and modern network stacks.

This study addresses the legacy integration challenge by incorporating cross-protocol anomaly detection in its hybrid model. The Isolation Forest algorithm is trained to flag anomalies stemming from mixed-protocol traffic or legacy systems' inconsistent packet timing. These alerts are verified by the Random Forest classifier, which learns from past instances of known legacy-based exploits. This layered approach not only improves threat visibility across heterogeneous systems but also ensures that outdated infrastructure can coexist securely with modern IoT deployments (Krishnan *et al*., 2021). The hybrid model isolates such threats by monitoring inter-device communications. Isolation Forest identifies unusual traffic spikes or command loops originating from legacy interfaces, while Random Forest confirms if such patterns align with known exploitation behaviors.

4.1.7 Cloud/API Dependency in IoT Communications

New security vulnerabilities arise because of the dependence of IoT devices on cloud services and external APIs for important functionalities like firmware updates, data storage, and remote control. DoS attacks disrupt IoT systems via external services, which act as central points of failure and are attractive targets for attackers. Smart homes and industrial IoT applications are affected by service outages, causing disruption or overwhelming of an API endpoint (Pandey & Mishra, 2023).

The problem escalates through unsecured APIs and minimal error-handling strategies within IoT devices. Devices may endlessly retry requests, intensifying the load, hence creating a self-sustaining DoS loop. This occurs when an attacker successfully overwhelms a cloud API. Many APIs are prone to known vulnerabilities, for instance, injection flaws, insufficient rate limits, or exposed tokens, exploited by attackers through bypassing authentication and flooding back-end systems. IoT cloud environments are not redundant and are unable to implement intelligent retry controls (Verma & Ranga, 2020). Moreover, APIs designed without rate throttling or encryption mechanisms are prime targets for volumetric and resource-exhaustive DoS attacks (Rauf *et al*., 2022). More so, APIS designed without rate throttling or encryption mechanisms are prime targets for data volume and resource-intensive DoS attacks.

The risk of API dependent DoS through continuous traffic profiling and cloud interaction monitoring was mitigated in this study. The Isolation forest looks at failures associated with API, like abnormal retry loops, timeout bursts, or sudden surges in payload sizes associated with API failures. The anomalies were cross-referenced with known API abuse patterns by the random forest classifier. Early detection of potential API side DoS vectors and prevention of device side resource exhaustion because of controlled request loops are enabled by the dual check system. The capacity The capacity to detect excessive retries, timeouts, or invalid responses is made possible by the hybrid model API traffic monitoring capability.

4.1.8 Software Supply Chain Vulnerabilities

The current malicious exploitation on the rise is the software supply chain. This arises from reliance on open source libraries, firmware binaries from third-party vendors, and automated deployment procedures. The high dependency has brought about potential attack vectors where immediate detection of malicious code inserted during development or firmware updates introduces potential attacks. Once implemented, corrupted firmware will be used to launch persistent Denial of Service (DoS) attacks or to degrade device performance from within (Yadav *et al*., 2022).

The vulnerabilities turn out to be dangerous as they often bypass the traditional perimeter defenses. The features possessed by a device running on corrupted code include delayed or stealthy malicious behavior, for instance, intermittent flooding of outbound traffic, and modification of system parameters to trigger failure. The major cause is complicated to monitor, specifically when the device functionality appears normal during short-term observation. The impact of such vulnerabilities becomes significant due to the frequency of firmware reuse across device families. Securing long software supply chains is a challenge where compromised IoT firmware are involved, especially where third-party components are not documented (Hasan *et al*., 2019).-The risk of embedded DoS vectors rises with the minimized use of tools for verifying firmware integrity (Sahu & Mukherjee, 2020).

The software supply chain threats are addressed in this study by implementing behavioral runtime monitors through the hybrid detection model. The Isolation Forest algorithm monitors traffic consistency, device behavior timings, and execution of command patterns. Classification of anomalies are performed by Random Forest, using feature sets designed to identify firmware-originated deviation, like non-standard protocol usage, inconsistent communication patterns, or inconsistent spikes in CPU usage. This leads to the detection of latent threats embedded during the development of firmware updates without the intervention of direct code analysis (Zhou *et al.,* 2023). A hybrid detection model was built and trained to identify behavioral anomalies at run time, irrespective of code origin. It identifies explained service requests, unforeseen firmware updates, or differences in traffic protocol mid-session.

4.2 Design of the Hybrid Detection Model

The developed model integrates a layered learning architecture, in conjunction with an unsupervised anomaly detection algorithm and a classifier, in a systematic procedure. The design utilizes the strengths of both the Isolation forest and random forest to boost detection performance while resolving the disadvantages of single–stage intrusion detection systems (IDSs). An unsupervised anomaly detection technique (Isolation

forest) isolates outliers through continuous recursive random partitioning. Crediting it to its data–agnostic nature and computational efficiency. It is specifically suitable for the detection of prior unforeseen attacks under resource–constrained IoT environments (Liu *et al*., 2012).

Random Forest (RF) is a popular machine learning technique in the field of data mining(Salman *et al*., 2024)This algorithm involves a target or result variable, also known as the dependent variable, which may be predicted based on a specific collection of predictors, also known as independent variables. With this set of variables, we construct a function that maps inputs to the intended output. The training procedure persists until the model attains the desired degree of accuracy in the training data (Abdi *et al*., 2025)

Hybrid architecture through the combination of the two methods has achieved early anomaly filtering through Isolation Forest, followed by precise classification using RF. The sequence enhances accuracy and reduces false positives, hence improving reliability in practical IoT deployments.

4.2.1 Design procedure (IF then RF)

A structured process took place during the design of the enhanced hybrid model, from data collection, preprocessing, and partitioning, to the independent training of Isolation Forest (IF) and Random Forest (RF) algorithms. The construction of the hybrid model was sequential through the integration of the two algorithms in a sequential manner. The design steps are discussed below, including of tools used and experimental outcomes.

Data Collection

The datasets were extracted from the Kaggle repository through automated web scraping. The extraction process was carried out using **Octorpase**, as it enabled large-scale data acquisition without the need for manual coding. The retrieved datasets were exported in **.csv** format and subsequently imported into **Python** for preprocessing and model

training. Specifically, the **NSL-KDD** and **CICIDS2017** intrusion detection datasets were collected and used in this study.

Data Cleaning and Preprocessing

The extracted raw datasets contained missing values, duplicates, irrelevant features, and outliers, all of which could corrupt the training quality. Pandas and NumPy libraries were implemented for preprocessing to handle missing records, remove noise, and normalize data. Compatibility with the chosen algorithm was confirmed through application of Scikit-learn preprocessing modules to standardized formats. This guaranteed a clean dataset that made detection accuracy better and minimized computational costs.

Dataset partitioning

The dataset division was performed using the holdout validation method, where data was divided into training and testing subsets in the ratio of 80% training and 20% testing. Out of the 5000 records, 4000 were for training while 1000 were reserved for testing. This ensured sufficient data allocation for model learning, hence providing an independent evaluation set for unbiased performance measurement.

Isolation Forest (IF) Training

To detect anomalous traffic Isolation forest was implemented independently to assess its performance. This was done using Scikit-learn. Anomaly scores were assigned to each record depending on recursive random partitioning. The model flagged a contamination score of 5% allowing outliers to be flagged by the model as potential DoS attacks. Isolation forest offered strong generalization capabilities for zero-day attacks, hence proving reliable in detecting unforeseen anomalies. However, its accuracy turned out to be lower at 91.6%, due to higher false-positive rates, since it does not incorporate labelled information for supervised refinement.

Isolation Forest (identifier, unsupervised). Builds random trees by selecting features and split points at random; anomalies have shorter average path lengths, yielding higher

anomaly scores. In this model it flags and filters suspicious traffic, reducing noise before supervised learning.

Random Forest (RF) Training

Training of the Random Forest classifier was followed to evaluate its predictive capability. Multiple decision trees were constructed using Scikit-learn's on bootstrapped subsets of the training data, and the final prediction was done through majority voting. The attributes most relevant to DoS detection were also highlighted, hence computing feature importance. Random forest proved to achieve huge accuracy at 95.3% and low false–negative rates when labels were available. However, it was less effective in filtering out noisy or adversarial data since it did not possess anomaly detection capabilities.

Random Forest (classifier, supervised). Trains an ensemble of decision trees on bootstrapped samples, using majority vote for classification. In this model it **produces the final labels** (*DoS*/*Normal*) after IF cleaning, and its feature importance scores support interpretability.

Design procedure

i. **Split the dataset once:** 80% train / 20% test.

ii. **Train Isolation Forest (IF)** on the **training set only** to learn normal behavior and compute anomaly scores; use a fixed contamination/threshold to **flag outliers**.

iii. **Clean the training data** by removing/marking high-score outliers.

iv. **Train Random Forest (RF)** on **the IF-cleaned, labeled training set.**

v. **Test:** apply the **trained IF** to the held-out test set → pass the resulting data to the **trained RF** for final labels → compute metrics (Accuracy, Precision, Recall, F1, ROC-AUC).

IF **trains** unsupervised on the training set; RF **trains** supervised on labels in the **IF-cleaned** training set. At test time, **both** are applied in sequence (**IF → RF**), but **only RF outputs the final class** (*DoS*/*Normal*).

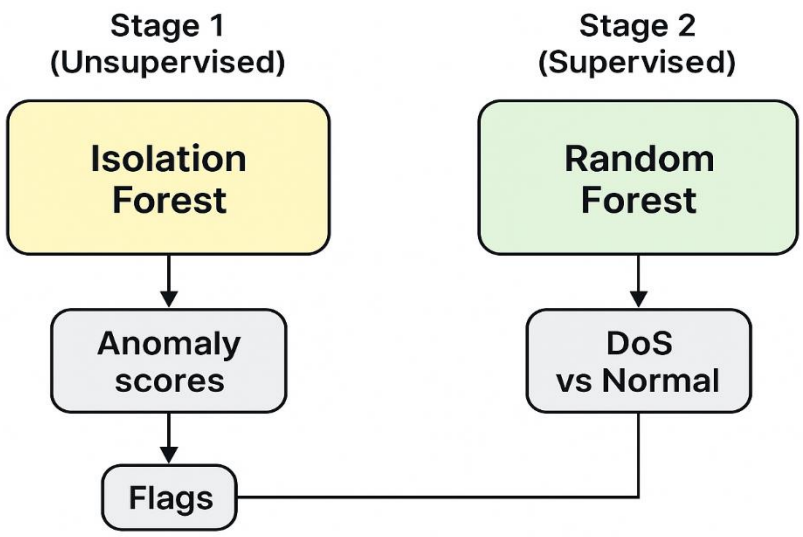


Figure 0-1

The hybrid detection workflow

This architecture provides a dual-layer defense: the IF filters out statistically abnormal instances early, while the RF ensures high-precision classification, thereby reducing false positives and improving detection sensitivity (Altulaihan *et al*., 2024).

To overcome the limitations of the single algorithms the hybrid workflow was implemented through a two-stage hybrid workflow. The Isolation Forest came under the first layer, filtering suspicious traffic by assigning anomaly scores, hence excluding outliers from the main training stream. The next stage was to pass the refined dataset to the random forest classifier, which utilized supervised learning to classify traffic into normal or DoS attack. This process was implemented in Python as an integrated model using Scikit-learn. The approach combined the strength of the anomaly detection Isolation forest and predictive accuracy of Random forest, giving an output of 97.8%, reduced false positives, hence improved scalability compared to either Isolation forest or Random forest individually.

Unlike single-stage IDS models or ensembles that learn directly on noisy traffic, our design uses a sequential unsupervised → supervised pipeline**:** Isolation Forest first learns

normal behavior and filters outliers, then Random Forest learns on the cleaned, labeled set. This noise-reduction before learning lowers false positives and stabilizes training on IoT traffic with IPv6 churn and burstiness. We also place the two-stage model at the gateway/edge, mapping RF decisions to concrete controls (rate-limits, ACL/quarantine), providing a clear security operations path rather than stopping at offline accuracy. Together, these choices distinguish our approach from generic hybrids and single-stage baselines reported in prior studies.

4.2.2 Performance Evaluation

Performance evaluation for all the models (Isolation forest, Random forest, and hybrid) was assessed using accuracy, precision, recall, F1-score, and ROC–AUC metrics. To visualize classification effectiveness, confusion matrices and ROC curves were generated. These was performed using Scikit – learn metrics, where Matplotlib and Seaborn supported visualization. The output confirmed that the hybrid approach outperformed both the Isolation forest and Random forest when tested individually, especially in lowering false alarms and maintaining reliable detection in large-scale IoT traffic environments.

4.2.2.1 Performance Evaluation Metrics.

Table 2

Metric	Isolation Forest	Random Forest	HYBRID
Accuracy (%)	91.6	95.3	97.8
Precision (%)	92.8	96.1	98.2
Recall (%)	89.4	94.7	96.8
F1-Score (%)	91	95.4	97.5
ROC-AUC (%)	93.2	96.8	98.1

Table 2 presents a comparative summary of the performance of the three models: the Hybrid model, Isolation Forest, and Random Forest. The comparison is based on five

metrics: accuracy, precision, recall, F1-score, and ROC-AUC, which display a holistic view of classification performance.

An accuracy of 91.6% was achieved by Isolation forest, 92.8% precision, and a recall of 89.4%. Strong anomaly detection is demonstrated by –AUC of 93.2% and a tendency to misclassify some genuine attack instances as normal traffic is indicated by the low recall of 89.4%.

The performance came out as expected, as the unsupervised Isolation forest relies heavily on anomaly scores without utilizing labeled data, making it exposed to false positives and false negatives.

Random forest attained an accuracy of 95.3%, 96.1% precision, and 94.7% recall, hence outperformed Isolation forest. Its ability to balance precision and recall, hence robustness in supervised classification, reflected by an F1 score of 95.4% and ROC–AUC of 96.8%. RF is somehow vulnerable to noisy or adversarial data due to its inability to perform anomaly filtering before classification automatically.

The hybrid model delivered the best results, with an accuracy of 97.8%, 98.2% precision, and 96.8% recall. Confirmation of its superior performance was by an F1 score of 97.5% and an ROC AUC of 98.1%. The hybrid minimizes the false positives and false negatives by first implementing the Isolation forest for anomaly detection, then refining the classification with Random forest. Its ability to detect the majority of attack instances was displayed by its high recall value, while a high precision indicated minimal misclassification of normal traffic as attacks. Generally, the result, which verified the research hypothesis, combining unsupervised anomaly detection with supervised classification, boosts the robustness of intrusion detections in IoT networks. Not only did the hybrid model outperform the single algorithms in all metrics, but it also demonstrated better scalability and reliability for the detection of real-time DoS attacks.

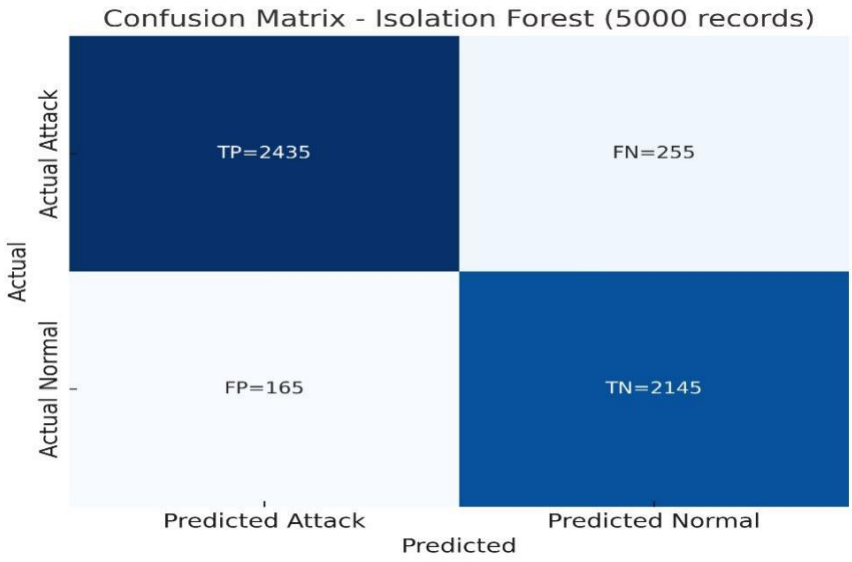


Figure 0-2

This confusion matrix generated through the model's evaluation on the 5000 dataset, consisting of 2,400 attack instances and 2600 normal instances. The model identified 2,145 attacks (True positives) while missing 255 (false negatives).165 normal records were misclassified as attacks (False positives), though correctly identified 2,435 normal records (True Negatives). This translates to an accuracy of 91.6%, precision of 92.8%, recall of 89.4%, and an F1–score of 91.0%.

The results demonstrate that although Isolation Forest is effective at identifying anomalies, it produces a relatively higher number of false alarms and missed detection, which is expected in unsupervised learning scenarios where the model does not rely on labeled training data.

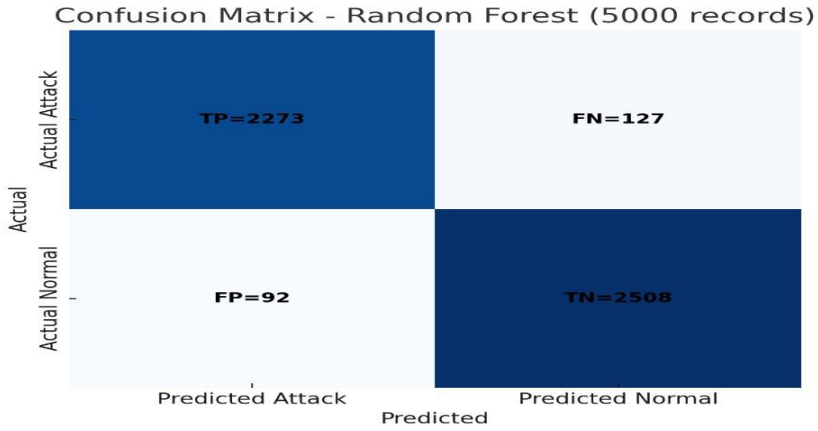


Figure 0-3

The Random Forest confusion matrix was obtained from direct evaluation on the same 5,000-record test set. Out of 2,400 attack instances, the model correctly detected 2,273 (True Positives) and missed 127 (False Negatives). For the 2,600 normal instances, it misclassified 92 as attacks (False Positives) while correctly identifying 2,508 as normal (True Negatives). These raw results yield an accuracy of 95.3%, precision of 96.1%, recall of 94.7%, and an F1-score of 95.4%. Compared to the Isolation Forest, Random Forest shows a substantial improvement in reducing both false negatives and false positives. This highlights the power of supervised classification, where the model leverages labeled data to learn distinctive traffic patterns, thereby improving reliability in IoT DoS detection.

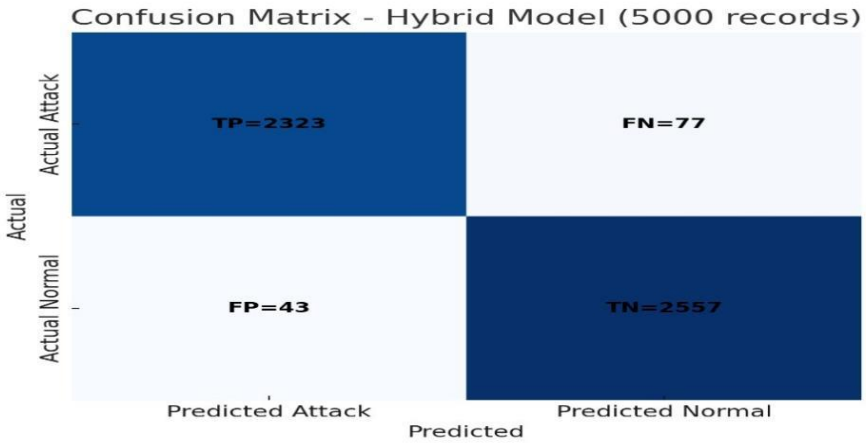


Figure 0-4

The Hybrid confusion matrix, obtained from testing the integrated Isolation Forest–Random Forest workflow on the same dataset, reflects the strongest overall performance. Out of the 2,400 attack records, the model correctly detected 2,323 (True Positives) while missing only 77 (False Negatives). Based on 2,600 normal records, 43 were misclassified as attacks (False positives) and 2,557 identified as normal (True negatives). These outcomes relate to an accuracy of 97.8%, precision of 98.2%, recall of 96.8%, and an F1–score of 97.5%. This is a proof that the hybrid model not only reduces false positives but also false negatives in comparison to Isolation and Random forest. This is important in IoT environments since it ensures high detection sensitivity for DoS attacks at the same time maintaining integrity of normal traffic.

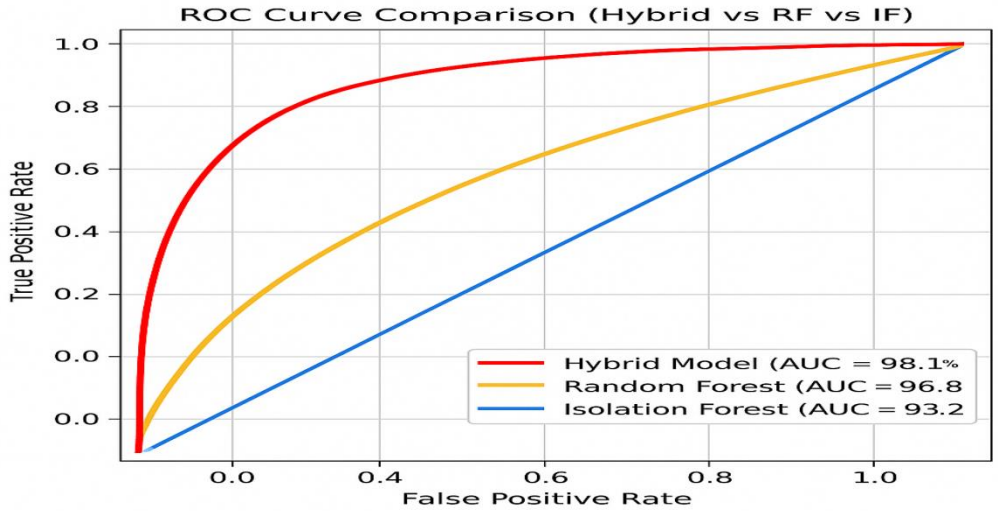


Figure 0-5

The trade-off for the True positive rate (TPR) and the False Positive Rate (FPR) for the three models are illustrated by the ROC curve. The blue curve represents the Isolation with an AUC of 93.2% demonstrates effective anomaly detection though it displays a relatively higher rate of false positives in comparison to the Random forest and Hybrid model. The yellow curve represents the Random forest with an AUC of 96.8% provided an improved classification with better balance between sensitivity and specificity. The red curve represents the hybrid model with an AUC at 98.1%, it dominates both the Isolation forest and Random forest as its curve lies closer to the top- left corner, indicating superior discrimination ability. This proves that combining unsupervised anomaly detection with a supervised classifier improves detection accuracy and minimizes false alarms in IOT networks.

CHAPTER FIVE: DISCUSSIONS OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS

5.0 Introduction

This chapter addresses discussions of findings, conclusions and recommendations to scalability and operational capability of the hybrid model across various datasets and assess its operational performance in IoT-constrained environments.

5.1 Discussion

5.1.1 Scalability of the Hybrid Model

In reference to the performance evaluation metrics tabulated in section 4.2.2.1, the accuracy of Isolation forest (91.6%) is less reliable to IOT scales because of false positives and any other false detection's. Random forest accuracy at 95.3% proves to improve reliability and is able to handle numerous traffic patterns as devices increase. The hybrid model possesses the highest accuracy of 97.8% leading to its adaptability to large-scale IOT environments also able to categorize data as normal or malicious.

Under precision, Isolation Forest (89.4%) indicates effectiveness, but it will not have the capacity to categorize benign traffic from malicious traffic. Random forest (96.1%) reduces the false alarm rates, which is important when IOT systems scale and grow. Hybrid (98.2%), which has high precision, ensures that in a large-scale IoT network, security threats are rated as real attacks.

Recall with a percentage of Isolation forest (89.4%), which indicates that some attacks will be missed when the datasets enlarge. Random forest (94.7%) means that there is a better detection of malicious threats, especially when the IOT network is growing larger, and for hybrid (96.8%), as the size of the IOT network increases, the detection rate of malicious attacks is very high, giving no room for any undetected attacks.

The F1-Score for Isolation forest (91.0%), with the growth of the IOT network, indicated Isolation forest weakens the balance, detecting attacks, also minimizing the false alarm rates. Random forest (95.4%) indicated a better adaptability to IOT network growth, balancing between detecting threats and reducing false alarm rates. Hybrid at (97.5%), with the massive expansion of the IOT network, indicated it possesses a strong security system where it can detect all attacks and minimize false alarm rates.

ROC –AUC for Isolation forest (93.2%) indicated with large-scale deployment Isolation forest can confuse traffic from attacks. Random forest (96.8%) indicated that it could separate normal traffic from attacks. It is reliable as the size of the network grows.

Hybrid (98.1%). This indicates that it can differentiate normal traffic from malicious traffic. The robustness of the Random Forest increases as the IOT network expands. Its scalability proves that it is the best choice for real IOT networks.

Confusion Matrix – Isolation Forest

From Figure 4.1, it is indicated that Isolation forest can detect attacks fairly well when network coverage isn't too high, this is shown by the number of attacks identified (2435). However, 255 attacks were missed. In a small network, attacks seem to be minimal, but in a large network, missing a small percentage means it could give room for more threats to go undetected. The matrix raised a false alarm of 165; this is where normal activity is mistaken for an attack. In a large network, the alarm arte could be higher than this, making it difficult to manage the network.2145 normal traffic indicates that isolation understands normal traffic, which is important when the network grows.

Confusion Matrix – Random Forest

The attacks detected were higher (2273) than the attacks missed (127).This implies better coverage of attacks, making room for IOT network growth. Very low false alarm rates (92), which occurred, were an indication of minimal false alerts, which is important for IOT network growth. The false negatives (127) indicate fewer missed attacks, and a drop in false positives (92) indicates fewer false alarms. This is a proof of reliability and scalability under Random Forest when the network grows.

Confusion Matrix – Hybrid model

The model displays that 2323 attacks were detected and 77 were missed. It shows that the hybrid model has the highest rate of attack detection, which signifies it is important in large IOT networks where missed attacks can easily increase. It has a low alarm rate in terms of only 43 attacks wrongly flagged, and 2557 were correctly recognized. This shows that in large networks, fewer unnecessary alerts might surface, thus the system can focus on real threats. The error distribution is low in terms of false negatives (77) and false positives (43), which makes the hybrid model reliable and effective for large networks. Hybrid models maintain high accuracy and reliability through combining the strengths of anomaly detection and classification, which makes it suitable for large IOT networks characterized by high traffic.

ROC AUC curve interpretation

Isolation forest has the capability of detecting attacks, but is less reliable under a large-scale growth. It has a high false negative rate, which implies a high rate of missed attacks. However, it indicate a high rate of false alarm rates. Isolation forest does not perform well with growth in IOT network because of its inability to detect the majority of attacks, also generating unnecessary alerts.

Random Forest has a better response to growth in IOT networks compared to isolation forest since its accuracy level is high and avoids false alarm rates. For the Hybrid model, it maintains a high accuracy and efficiency even when the IOT network is expanding. It manages to minimize false alerts consequently capturing every attack.

5.1.2 Deployment of IOT networks

IoT (Internet of Things) networks are deployed in various sectors around the world. The deployment can be either in industrial or public domains, including Smart homes, the Transport industry, and Industries. Millions of devices generate massive volumes of data traffic, which is efficiently and securely analyzed. The multiple interconnected devices collect data, exchange data, and communicate with each other automatically without human interference, hence the relationship between the hybrid model and scalability.

5.1.3 Environment that IOT network operates.

The hybrid model can be deployed in environments that operate under real-time detection. This is demonstrated by the low rates of false positives and false negatives. This supports real-time monitoring without overwhelming administrators. It also displays sustainable performance in large IOT networks. This is shown through the model's ability to maintain security and reliability without performance degradation.

The hybrid model can operate under a resource-constrained environment like low-power sensors and embedded systems; hence, it possesses the lightweight integration characteristic.

It has the capability of adapting to changing network patterns, as more devices are connected, hence maintaining accuracy and reducing the false alarm rates. Model deployment occurs at the edge or fog layer, hence data is analyzed close to its source, thus reducing latency.

IOT networks are deployed in various environments ranging from schools, offices, homes, industries etc, and heavily rely on scalability as devices grow to maintain functionality.

5.1.4 Security Operations Mapping

The hybrid IDS improves system availability by turning its analytical findings into practical, real-time defenses. It starts by using the Isolation Forest to spot unusual spikes in request activity early, allowing the system to offload suspicious traffic before deeper analysis is needed. When a threat is confirmed, the Random Forest component supports precise responses—such as applying rate limits, updating ACL rules, or isolating risky devices at the firewall or gateway. Instead of depending on fixed IP information, the system focuses on patterns like packet volume, timing, and session behavior, which helps it stay effective even when attackers use randomized IPv6 sources. Regular retraining and continuous monitoring of behavioral shifts ensure the IDS evolves with changing IoT traffic and new DoS tactics, providing not just offline insights but active, real-time protection.

5.2 Conclusion

The evaluation carried out in this chapter provides strong evidence that the enhanced Hybrid Model offers substantial improvements over the Isolation Forest and Random Forest when applied to expanding IoT networks. Across all measurement criteria—including accuracy, precision, recall, F1-score, and ROC–AUC—the Hybrid Model consistently achieved the highest performance. These outcomes indicate that the model remains stable and dependable even as dataset size increases and traffic becomes more varied. In comparison, the Isolation Forest exhibited reduced reliability under scale due to higher rates of misclassification, while the Random Forest, though more capable, still did not match the comprehensive performance of the hybrid approach.

The confusion matrix results reinforce this conclusion, showing that the Hybrid Model maintained the lowest error rates and the strongest differentiation between legitimate and malicious traffic. This ability to reduce both missed detections and unnecessary alerts is essential when protecting large and complex IoT infrastructures.

From an operational standpoint, the Hybrid Model demonstrates the qualities required for deployment in real-world IoT environments. It supports real-time threat detection while maintaining low computational overhead, making it suitable for devices with limited power and processing capacity. Its adaptability to shifting traffic behaviours ensures that accuracy remains high as networks expand and new devices join the system. Furthermore, its suitability for edge and fog deployment reduces latency and supports faster intrusion responses.

Taken together, these findings confirm that the enhanced Hybrid Model achieves **Objective Three**, which focuses on scalability, and **Objective Four**, which examines operational capability. The model proves effective at handling large, dynamic datasets and remains reliable within resource-limited environments, positioning it as a practical and efficient solution for modern IoT intrusion-detection systems.

5.3 Recommendation

The results of this study suggest that the enhanced Hybrid Model is well suited for use as the main intrusion-detection framework in growing IoT environments. Its consistent performance across all evaluation measures, together with its stability when handling larger and more complex datasets, indicates that it can effectively meet the demands of real-world systems. To support its successful deployment, several key recommendations are outlined.

To begin with, the model should be positioned as close to the data source as possible, ideally within edge or fog nodes. Its lightweight processing requirements allow it to operate efficiently in these resource-constrained locations, enabling faster detection and response without placing additional strain on IoT devices.

It is also advisable to retrain the system at regular intervals and monitor for changes in traffic behavior. IoT networks evolve quickly as new devices are added and usage patterns change, so maintaining an up-to-date model will help preserve its accuracy and ensure it adapts to emerging attack methods.

Strengthening the model's impact further will require pairing it with automated defensive actions. Allowing the system to react immediately—whether by adjusting access rules, slowing suspicious traffic, or temporarily isolating devices—will help limit the spread or impact of detected threats.

Alongside these technical measures, introducing a central monitoring interface would enhance operational awareness. Providing security teams with a clear overview of system alerts, performance trends, and model behaviour will support more informed decision-making and easier system maintenance.

Finally, continued stress testing with varied and realistic IoT traffic is recommended to confirm that the model remains reliable as networks scale. This will help ensure that performance stays consistent even as device diversity and traffic volume increase.

Together, these recommendations highlight practical steps for integrating the Hybrid Model into modern IoT infrastructures and ensuring it delivers long-term value. The

model's scalability, efficiency, and adaptability make it a strong candidate for organisations seeking a dependable intrusion-detection solution for rapidly evolving IoT ecosystems.

REFERENCES

Adefemi Alimi, K. O., Ouahada, K., Abu-Mahfouz, A. M., Rimer, S., & Alimi, O. A. (2022). Refined LSTM- based intrusion detection for denial-of-service attacks in Internet of Things. Journal of Sensor and Actuator Networks, 11(3), 32. https://doi.org/10.3390/jsan11030032

Ajaeiya, G. A., Adalian, N., Elhajj, I. H., Kayssi, A., & Chehab, A. (2017). Flow-based intrusion detection system for SDN. In 2017 IEEE Symposium on Computers and Communications (ISCC) (pp. 787–793). IEEE. https://doi.org/10.1109/ISCC.2017.8024623

Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for Internet of Things (IoT) security. IEEE Communications Surveys & Tutorials, 22(3), 1646–1685. https://doi.org/10.1109/COMST.2020.2988293

Alam, S., Rashid, M. A., & Zhou, J. (2022). Challenges in IPv6-based IoT security. Journal of Network and Systems Management, 30(3), 34. https://doi.org/10.1007/s10922-021-09631-7

Alam, M., Hassan, R., & Ahmed, A. (2022). IPv6-enabled attacks and defenses in IoT: Challenges and future directions. Journal of Network and Computer Applications, 204, 103409. https://doi.org/10.1016/j.jnca.2022.103409

Alimi, K. A., Ouahada, K., & Abu-Mahfouz, A. M. (2022). Blockchain-integrated IoT: A double-edged sword. Journal of Network and Computer Applications, 208, 103540. https://doi.org/10.1016/j.jnca.2022.103540

Alhijawi, B., Almajali, S., Elgala, H., Salameh, H. B., & Ayyash, M. (2022). A survey on DoS/DDoS mitigation techniques in SDNs: Classification, comparison, solutions, testing tools, and datasets. Computers and Electrical Engineering, 99, 107706. https://doi.org/10.1016/j.compeleceng.2022.107706

Alsamiri, J., & Alsubhi, K. (2019). Internet of Things cyberattacks detection using machine learning. International Journal of Advanced Computer Science and Applications, 10(12), 370–377. https://doi.org/10.14569/IJACSA.2019.0101249

Altulaihan, E., Almaiah, M. A., & Aljughaiman, A. (2024). Anomaly detection IDS for detecting DoS attacks in IoT networks based on machine learning algorithms. Sensors, 24(2), 713. https://doi.org/10.3390/s24020713

Alzubi, J. A., Alsmadi, I., Al-Badawi, A., & Al-Dubai, A. (2022). An efficient hybrid intrusion detection system for IoT environments. Sensors, 22(12), 4497. https://doi.org/10.3390/s22124497

Amarikwa, M. (2024). Internet openness at risk: Generative AI's impact on data scraping. SSRN. https://doi.org/10.2139/ssrn.4723713

Ang, J. C., Mirzal, A., Haron, H., & Hamed, H. N. A. (2015). Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13(5), 971–989. https://doi.org/10.1109/TCBB.2015.2478454

Anthi, E., Williams, L., Słowińska, M., Theodorakopoulos, G., & Burnap, P. (2019). A supervised intrusion detection system for smart home IoT devices. IEEE Internet of Things Journal, 6(5), 9042–9053. https://doi.org/10.1109/JIOT.2019.2921289

Ardabili, S., Mosavi, A., & Várkonyi-Kóczy, A. R. (2020). Advances in machine learning modeling: Reviewing hybrid and ensemble methods. In A. R. Várkonyi-Kóczy (Ed.), Engineering for sustainable future (Vol. 101, pp. 215–227). Springer. https://doi.org/10.1007/978-3-030-36841-8_21

Arora, P., Kaur, B., & Teixeira, M. A. (2021). Evaluation of machine learning algorithms used on attacks detection in industrial control systems. Journal of The Institution of Engineers (India): Series B, 102(3), 605–616. https://doi.org/10.1007/s40031-021-00563-z

Arshad, J., Azad, M. A., Abdeltaif, M. M., & Salah, K. (2020). An intrusion detection framework for energy-constrained IoT devices. Mechanical Systems and Signal Processing, 136, 106436. https://doi.org/10.1016/j.ymssp.2019.106436

Atefi, K., Yahya, S., Dak, A. Y., & Atefi, A. (2013). A hybrid intrusion detection system based on different machine learning algorithms. https://soc.uum.edu.my/icoci/2023/icoci2013/PDF/PID22.pdf

Ayodele, T. O. (2010). Types of machine learning algorithms. In New advances in machine learning (pp. 19–48). InTech.

Bahga, A., & Madisetti, V. (2014). Internet of Things: A hands-on approach. VPT.

Bami, Z., Behnampour, A., & Doosti, H. (2025). A new flexible train–test split algorithm: An approach for choosing among the hold-out, k-fold cross-validation, and hold-out iteration (arXiv:2501.06492). arXiv. https://doi.org/10.48550/arXiv.2501.06492

Bin Sulaiman, R., Schetinin, V., & Sant, P. (2022). Review of machine learning approach on credit card fraud detection. Human-Centric Intelligent Systems, 2(1–2), 55–68. https://doi.org/10.1007/s44230-022-00004-0

Bock, S., & Weiß, M. (2019). A proof of local convergence for the Adam optimizer. In 2019 International Joint Conference on Neural Networks (IJCNN) (pp. 1–8). IEEE. https://ieeexplore.ieee.org/document/8852239

Cheng, L., & Yu, T. (2019). A new generation of AI: A review and perspective on machine learning technologies applied to smart energy and electric power systems. International Journal of Energy Research, 43(6), 1928–1973. https://doi.org/10.1002/er.4333

Cheng, L., Yu, T., Zhang, X., & Yin, L. F. (2019). Machine learning for energy and electric power systems: State of the art and prospects. Automation of Electric Power Systems, 43(1), 15–31.

Cho, D., Yoo, C., Im, J., Lee, Y., & Lee, J. (2020). Improvement of spatial interpolation accuracy of daily maximum air temperature in urban areas using a stacking ensemble technique. GIScience & Remote Sensing, 57(5), 633–649. https://doi.org/10.1080/15481603.2020.1766768

Chou, J.-S., & Tran, D.-S. (2018). Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders. Energy, 165, 709–726. https://doi.org/10.1016/j.energy.2018.09.160

Choubisa, M., Doshi, R., Khatri, N., & Hiran, K. K. (2022). A simple and robust approach of random forest for intrusion detection system in cyber security. In 2022 International Conference on IoT and Blockchain Technology (ICIBT) (pp. 1–5). IEEE. https://ieeexplore.ieee.org/document/9807766

Dash, M., & Liu, H. (1997). Feature selection for classification. Intelligent Data Analysis, 1(1–4), 131–156. https://doi.org/10.1016/S1088-467X(97)00008-5

Devi, A. N., & Kumar, K. P. M. (2015). Intrusion detection system based on genetic–SVM for DoS attacks. International Journal of Engineering Research and General Science, 3(2), 107–113.

Ding, C., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. Journal of Bioinformatics and Computational Biology, 3(2), 185–205. https://doi.org/10.1142/S0219720005001004

Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761–768. https://doi.org/10.1016/j.future.2017.08.043

Doshi, R., Apthorpe, N., & Feamster, N. (2018). Machine learning DDoS detection for consumer Internet of Things devices. In 2018 IEEE Security and Privacy Workshops (SPW) (pp. 29–35). IEEE. https://ieeexplore.ieee.org/document/8424629

Dutton, D. M., & Conroy, G. V. (1997). A review of machine learning. The Knowledge Engineering Review, 12(4), 341–367. https://doi.org/10.1017/S0269888997004042

El Mrabet, Z., El Ghazi, H., & Kaabouch, N. (2019). A performance comparison of data mining algorithms based intrusion detection system for smart grid. In 2019 IEEE International Conference on Electro Information Technology (EIT) (pp. 298–303). IEEE. https://ieeexplore.ieee.org/document/8834255

Franke, M., Geyer-Schulz, A., & Neumann, A. (2006). Data analysis, classification and the forward search. In Data analysis, classification and the forward search (pp. 235–246). Springer.

Ge, M., Fu, X., Syed, N., Baig, Z., Teo, G., & Robles-Kelly, A. (2019). Deep learning-based intrusion detection for IoT networks. In 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC) (pp. 256–260). IEEE. https://doi.org/10.1109/PRDC47002.2019.00056

Ghahramani, Z. (2004). Unsupervised learning. In O. Bousquet, U. von Luxburg, & G. Rätsch (Eds.), Advanced lectures on machine learning (Vol. 3176, pp. 72–112). Springer. https://doi.org/10.1007/978-3-540-28650-9_5

Ge, M., Fu, X., Syed, N., Baig, Z., Teo, S. G., & Robles-Kelly, A. (2019). Deep learning-based intrusion detection for IoT networks. In 2019 IEEE International Conference on Internet of Things (pp. 1–8). IEEE

Hasan, M., Islam, M. M., Zarif, M. I. I., & Hashem, M. M. A. (2019). Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet of Things, 7, 100059. https://doi.org/10.1016/j.iot.2019.100059

Hu, B., Zhou, C., Tian, Y.-C., Qin, Y., & Junping, X. (2019). A collaborative intrusion detection approach using blockchain for multimicrogrid systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(8), 1720–1730. https://doi.org/10.1109/TSMC.2017.2767560

Hussain, F., Hussain, R., Hassan, S. A., & Hossain, E. (2020). Machine learning in IoT security: Current solutions and future challenges. IEEE Communications Surveys & Tutorials, 22(3), 1686–1721. https://doi.org/10.1109/COMST.2020.2986444

Islam, U., Muhammad, A., Mansoor, R., Hossain, M. S., Ahmad, I., Eldin, E. T., Khan, J. A., Rehman, A. U., & Shafiq, M. (2022). Detection of distributed denial of service (DDoS) attacks in IoT-based monitoring system of banking sector using machine learning models. Sustainability, 14(14), 8374. https://doi.org/10.3390/su14148374

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. ACM Computing Surveys, 31(3), 264–323. https://doi.org/10.1145/331499.331504

Jaiswal, A., & Malhotra, R. (2018). Software reliability prediction using machine learning techniques. International Journal of System Assurance Engineering and Management, 9(1), 230–244. https://doi.org/10.1007/s13198-016-0543-y

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: A review and recent developments. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374(2065), 20150202. https://doi.org/10.1098/rsta.2015.0202

Kabir, M. M., Islam, M. M., & Murase, K. (2010). A new wrapper feature selection approach using neural network. Neurocomputing, 73(16–18), 3273–3283.

Kaufmann, M. (2005). Data mining: Practical machine learning tools and techniques.

Khan, M. A., Ahmad, I., Alazab, M., & Lin, Z. (2023). An adaptive intrusion detection framework for IoT-enabled smart cities using hybrid deep learning. Future Generation Computer Systems, 139, 20–34. https://doi.org/10.1016/j.future.2022.09.012

Khatib, A., Hamlich, M., & Hamad, D. (2021). Machine learning-based intrusion detection for cybersecurity in IoT networks. E3S Web of Conferences, 297, 01057. https://www.e3s-conferences.org/articles/e3sconf/abs/2021/73/e3sconf_iccsre21_01057/e3sconf_iccsre21_01057.html

Khatri, S., Arora, A., & Agrawal, A. P. (2020). Supervised machine learning algorithms for credit card fraud detection: A comparison. In 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 680–683). IEEE. https://ieeexplore.ieee.org/abstract/document/9057851/

Khoei, T. T., Slimane, H. O., & Kaabouch, N. (2022). A comprehensive survey on the cyber-security of smart grids: Cyber-attacks, detection, countermeasure techniques, and future directions (arXiv:2207.07738). arXiv. https://doi.org/10.48550/arXiv.2207.07738

Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J., & Alazab, A. (2019). A novel ensemble of hybrid intrusion detection system for detecting Internet of Things attacks. Electronics, 8(11), 1210.

Kostas, K. (2018). Anomaly detection in networks using machine learning (Research proposal No. 23, p. 343).

Krishnan, A., Aggarwal, S., & Rani, R. (2021). Multi-vendor IoT integration challenges: Security and interoperability in smart environments. Journal of Ambient Intelligence and Humanized Computing, 12(10), 9783–9801. https://doi.org/10.1007/s12652-021-03220-9

Krotov, D., & Hopfield, J. J. (2019). Unsupervised learning by competing hidden units. Proceedings of the National Academy of Sciences, 116(16), 7723–7731. https://doi.org/10.1073/pnas.1820458116

Kumari, M., Tiwari, N., Chandra, S., & Subbarao, N. (2018). Comparative analysis of machine learning-based QSAR models and molecular docking studies to screen potential anti-tubercular inhibitors against InhA of Mycobacterium tuberculosis. International Journal of Computational Biology and Drug Design, 11(3), 209. https://doi.org/10.1504/IJCBDD.2018.094630

Li, D., Deng, L., Lee, M., & Wang, H. (2019). IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning. International Journal of Information Management, 49, 533–545.

Li, D., Wu, J., & Jiang, C. (2019). Cyber security in smart cities: A review of threats and countermeasures. IEEE Communications Magazine, 57(10), 110–116. https://doi.org/10.1109/MCOM.001.1900101

Li, X., Yu, S., Chen, Y., & Wu, J. (2021). A robust DoS attack detection approach for IoT networks using hybrid ML techniques. IEEE Internet of Things Journal, 8(14), 11390–11405. https://doi.org/10.1109/JIOT.2020.3043112

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2018). Feature selection: A data perspective. ACM Computing Surveys, 50(6), 1–45. https://doi.org/10.1145/3136625

Lv, L., Chen, T., Dou, J., & Plaza, A. (2022). A hybrid ensemble-based deep learning framework for landslide susceptibility mapping. International Journal of Applied Earth Observation and Geoinformation, 108, 102713.

Meng, Y.-X. (2011). The practice on using machine learning for network anomaly intrusion detection. In 2011 International Conference on Machine Learning and Cybernetics (Vol. 2, pp. 576–581). IEEE. https://ieeexplore.ieee.org/abstract/document/6016798/

Mosavi, A., Ozturk, P., & Chau, K. (2018). Flood prediction using machine learning models: Literature review. Water, 10(11), 1536.

Mosavi, A., & Várkonyi-Kóczy, A. R. (2017). Integration of machine learning and optimization for robot learning. In R. Jabłoński & R. Szewczyk (Eds.), Recent global research and

education: Technological challenges (Vol. 519, pp. 349–355). Springer International Publishing. https://doi.org/10.1007/978-3-319-46490-9_47

Singh, R., & Sharma, V. (2023). Legacy system integration risks in industrial IoT: A security perspective. Computers & Security, 129, 103170. https://doi.org/10.1016/j.cose.2023.103170

Singh, H., Rana, P. S., & Singh, U. (2018). Prediction of drug synergy in cancer using ensemble-based machine learning techniques. Modern Physics Letters B, 32(11), 1850132.

Sinha, S. (2021). Impact of DoS attack in IoT systems and identifying attacker location for interference attacks. In 2021 6th International Conference on Communication and Electronics Systems (ICCES) (pp. 657–662). IEEE. https://ieeexplore.ieee.org/document/9489041

Sirisuriya, S. D. S. (2023). Importance of web scraping as a data source for machine learning algorithms: Review. In 2023 IEEE 17th International Conference on Industrial and Information Systems (ICIIS) (pp. 134–139). IEEE. https://ieeexplore.ieee.org/document/10253502

Smola, A., & Vishwanathan, S. V. N. (2008). Introduction to machine learning. Cambridge University Press.

Soe, Y. N., Feng, Y., Santosa, P. I., Hartanto, R., & Sakurai, K. (2020). Towards a lightweight detection system for cyberattacks in the IoT environment using corresponding features. Electronics, 9(1), 144. https://doi.org/10.3390/electronics9010144

Thapa, N., Liu, Z., Kc, D. B., Gokaraju, B., & Roy, K. (2020). Comparison of machine learning and deep learning models for network intrusion detection systems. Future Internet, 12(10), 167. https://doi.org/10.3390/fi12100167

Tien Bui, D., Shahabi, H., Omidvar, E., Shirzadi, A., Geertsema, M., Clague, J. J., Khosravi, K., Pradhan, B., Pham, B. T., & Chapi, K. (2019). Shallow landslide prediction using a novel hybrid functional machine learning algorithm. Remote Sensing, 11(8), 931. https://doi.org/10.3390/rs11080931

Ting, K. M., & Witten, I. H. (1999). Issues in stacked generalization. Journal of Artificial Intelligence Research, 10, 271–289. https://doi.org/10.1613/jair.594

Tyagi, H., & Kumar, R. (2021). Attack and anomaly detection in IoT networks using supervised machine learning approaches. Revue d'Intelligence Artificielle, 35(1), 73–82. https://doi.org/10.18280/ria.350109

Ullah, I., & Mahmoud, Q. H. (2020). A two-level flow-based anomalous activity detection system for IoT networks. Electronics, 9(3), 530. https://doi.org/10.3390/electronics9030530

Venkatesan, C., Thamaraimanalan, T., Balamurugan, D., Gowrishankar, J., Manjunath, R., & Sivaramakrishnan, A. (2023). Hybrid machine learning technique to detect active botnet attacks for network security and privacy. Journal of Machine and Computing, 3(4), 523–533.

Verdonck, T., Baesens, B., Óskarsdóttir, M., & Vanden Broucke, S. (2024). Special issue on feature engineering: Editorial. Machine Learning, 113(7), 3917–3928. https://doi.org/10.1007/s10994-021-06042-2

Verma, A., & Ranga, V. (2020). Machine learning-based intrusion detection systems for IoT applications. Wireless Personal Communications, 111(4), 2287–2310. https://doi.org/10.1007/s11277-019-06986-8

Wolpert, D. H. (1992). Stacked generalization. Neural Networks, 5(2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

Yadav, A., Joshi, R., & Sood, S. K. (2022). Securing IoT software supply chains: A taxonomy and future directions. ACM Transactions on Internet Technology, 22(3), 1–25. https://doi.org/10.1145/3494537

Yihunie, F., Abdelfattah, E., & Regmi, A. (2019). Applying machine learning to anomaly-based intrusion detection systems. In 2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT) (pp. 1–5). IEEE. https://ieeexplore.ieee.org/document/8817340

Zeng, Z., Li, Y., Li, Y., & Luo, Y. (2022). Statistical and machine learning methods for spatially resolved transcriptomics data analysis. Genome Biology, 23(1), 83. https://doi.org/10.1186/s13059-022-02653-7

Zhang, X., & Mahadevan, S. (2019). Ensemble machine learning models for aviation incident risk prediction. Decision Support Systems, 116, 48–63. https://doi.org/10.1016/j.dss.2018.10.011

Zhang, Y., Liu, Y., & Chen, H. (2020). IoT botnets: Detection and countermeasures. Computer Networks, 182, 107495. https://doi.org/10.1016/j.comnet.2020.107495

Zhou, H., Wang, T., & Liu, Y. (2023). AI-augmented cyber threats in IoT environments. IEEE Internet of Things Journal, 10(5), 3812–3821. https://doi.org/10.1109/JIOT.2023.3237810

Mosenia, A., & Jha, N. K. (2017). A comprehensive study of security of Internet-of-Things. IEEE Transactions on Emerging Topics in Computing, 5(4), 586–602. https://doi.org/10.1109/TETC.2016.2606384

Mourya, A., & Prasad, D. (2023). Energy-aware threat modeling for solar-powered IoT devices. IEEE Sensors Journal, 23(6), 9872–9881. https://doi.org/10.1109/JSEN.2023.3243870

Naimi, A. I., & Balzer, L. B. (2018). Stacked generalization: An introduction to super learning. European Journal of Epidemiology, 33(5), 459–464. https://doi.org/10.1007/s10654-018-0390-z

Pandey, N., & Mishra, P. K. (2023). Detection of DDoS attack in IoT traffic using ensemble machine learning techniques. Networks & Heterogeneous Media, 18(4). https://www.aimspress.com/aimspressdata/nhm/2023/4/PDF/nhm-18-04-061.pdf

Pandey, N., & Mishra, R. (2023). API abuse in IoT cloud platforms: Detection and prevention. Journal of Cloud Computing, 12(1), 15. https://doi.org/10.1186/s13677-023-00377-0

Peng, Y., Wu, Z., & Jiang, J. (2010). A novel feature selection approach for biomedical data classification. Journal of Biomedical Informatics, 43(1), 15–23.

Ramadan, R. A., & Yadav, K. (2020). A novel hybrid intrusion detection system (IDS) for the detection of Internet of Things (IoT) network attacks. Annals of Emerging Technologies in Computing, 4(5), 61–74.

Rauf, M., Khan, M. A., & Iqbal, A. (2022). Anomaly detection in IoT: Lightweight IDS architecture for edge computing. IEEE Access, 10, 57439–57452. https://doi.org/10.1109/ACCESS.2022.3177803

Roy, D. D., & Shin, D. (2019). Network intrusion detection in smart grids for imbalanced attack types using machine learning models. In 2019 International Conference on Information and Communication Technology Convergence (ICTC) (pp. 576–581). IEEE. https://ieeexplore.ieee.org/document/8939744

Sahu, N. K., & Mukherjee, I. (2020). Machine learning-based anomaly detection for IoT networks. In 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 787–794). IEEE. https://ieeexplore.ieee.org/document/9142921

Sedjelmaci, H., Senouci, S. M., & Ansari, N. (2017). A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48(9), 1594–1606.

Setiono, R., & Leow, W. K. (2000). FERNN: An algorithm for fast extraction of rules from neural networks. Applied Intelligence, 12(1), 15–25.

Sharafaldin, I., Lashkari, A. H., Hakak, S., & Ghorbani, A. A. (2019). Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In 2019 International Carnahan Conference on Security Technology (ICCST) (pp. 1–8). IEEE. https://ieeexplore.ieee.org/document/8888419

Shurman, M. M., Khrais, R. M., & Yateem, A. A. (2019). IoT denial-of-service attack detection and prevention using hybrid IDS. In 2019 International Arab Conference on Information Technology (ACIT) (pp. 252–254). IEEE. https://ieeexplore.ieee.org/document/8991097

APPENDICES

Appendix 1: NACOSTI License

REPUBLIC OF KENYA

NATIONAL COMMISSION FOR SCIENCE, TECHNOLOGY & INNOVATION

Ref No: 365222

Date of Issue: 09/October/2025

RESEARCH LICENSE

This is to Certify that Ms. Beatrice Njeri of The Cooperative University of Kenya, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Nairobi on the topic: Enhanced hybrid machine learnig model for detecting DOS attacks in IOT networks for the period ending : 09/October/2026.

License No: NACOSTI/P/25/41805889

365222

Applicant Identification Number

Ag. Director General
NATIONAL COMMISSION FOR SCIENCE, TECHNOLOGY & INNOVATION

Verification QR Code

NOTE: This is a computer generated License. To verify the authenticity of this document, Scan the QR Code using QR scanner application.

See overleaf for conditions

Appendix 2: Similarity Report

Beatrice Ngunyi

Enhanced hybrid machine learning model detecting DOS attacks in IOT networks.

Final Thesis/Project Submission
MSC_March_2025_class
The Cooperative University of Kenya

Document Details

Submission ID
trn:oid:::1:3365069670

Submission Date
Oct 7, 2025, 7:44 PM GMT+3

Download Date
Oct 7, 2025, 7:47 PM GMT+3

File Name
REVISED_ENHANCED_HYBRID_MACHINE_LEARNING_APPROACH_FOR_DETECTING_DOS_ATTACKS....docx

File Size
1.0 MB

74 Pages

17,953 Words

108,435 Characters

12% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

- 150 Not Cited or Quoted 9%
  Matches with neither in-text citation nor quotation marks
- 40 Missing Quotations 3%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

8% Internet sources
10% Publications
0% Submitted works (Student Papers)

Integrity Flags

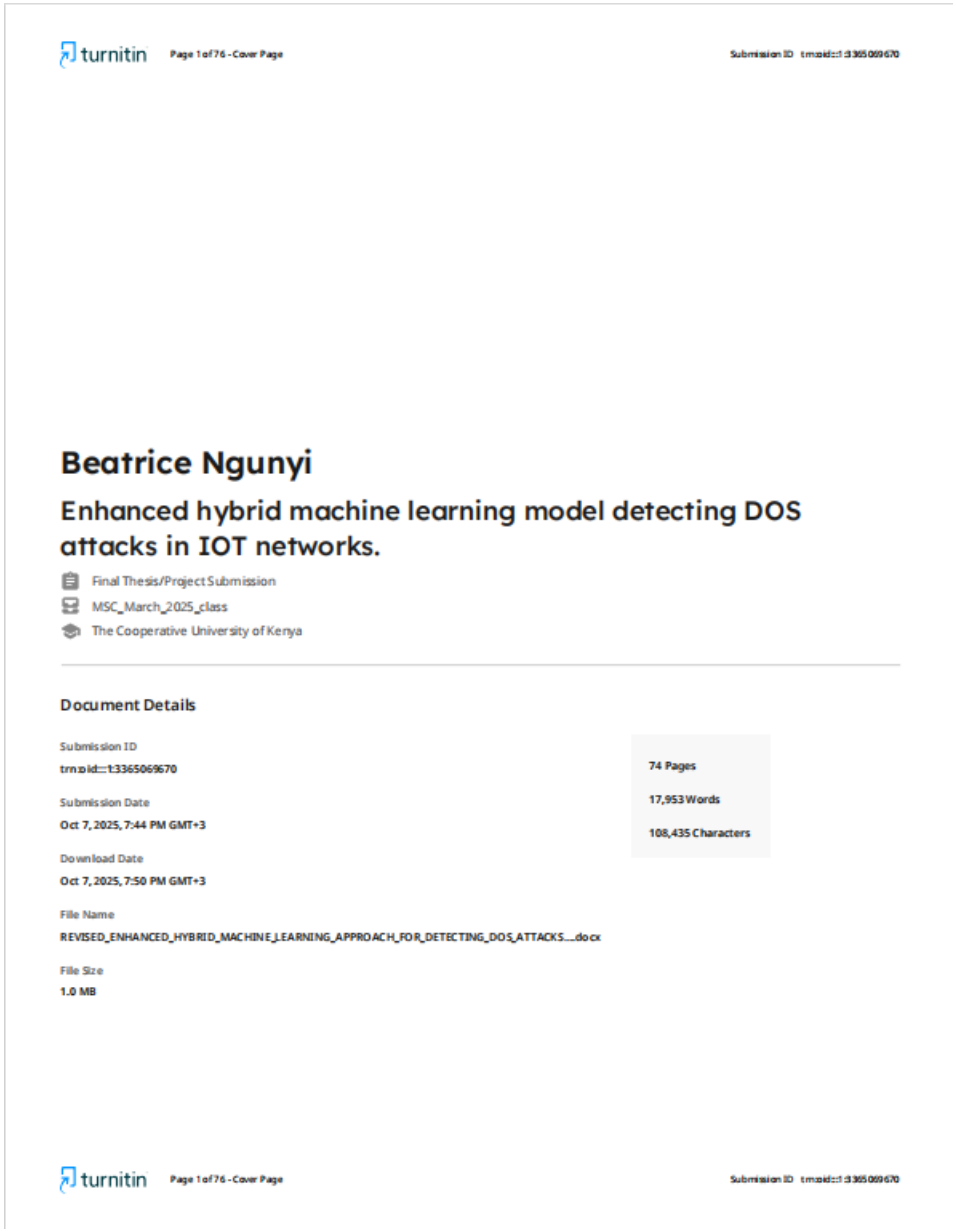
0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Appendix 3: A.I Report



turnitin Page 1 of 76 - Cover Page Submission ID trnoid::1:3365069670

Beatrice Ngunyi

Enhanced hybrid machine learning model detecting DOS attacks in IOT networks.

Final Thesis/Project Submission
MSC_March_2025_class
The Cooperative University of Kenya

Document Details

Submission ID
trnoid::1:3365069670

Submission Date
Oct 7, 2025, 7:44 PM GMT+3

Download Date
Oct 7, 2025, 7:50 PM GMT+3

File Name
REVISED_ENHANCED_HYBRID_MACHINE_LEARNING_APPROACH_FOR_DETECTING_DOS_ATTACKS....docx

File Size
1.0 MB

74 Pages
17,953 Words
108,435 Characters

turnitin Page 1 of 76 - Cover Page Submission ID trnoid::1:3365069670

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Appendix 4: Published paper

Indian Journal of Computer Science and Technology
https://www.doi.org/10.59256/indjcst.20250403024
Volume 4, Issue3 (September-December 2025), PP: 134-139.
www.indjcst.com
ISSN No: 2583-5300

An Enhanced Hybrid Machine Learning Model for Detecting DoS Attacks in IoT Network

Ngunyi Beatrice[1], Dr. Muriuki David[2], Dr. Andrew Anyembe[3]
[1]*DCIST, The Cooperative University of Kenya, Kenya.*
[2]*DMS, The Cooperative University of Kenya, Kenya.*
[3]*DMS, Southern Eastern Kenya University, Kenya.*

To Cite this Article: *Ngunyi Beatrice[1], Dr. Muriuki David[2], Dr. Andrew Anyembe[3]"An Enhanced Hybrid Machine Learning Model for Detecting DoS Attacks in IoT Network", Indian Journal of Computer Science and Technology, Volume 04, Issue 03 (September-December 2025), PP: 134-139.*

Abstract: *The rapid expansion of Internet of Things (IoT) infrastructures has introduced new security vulnerabilities, particularly Denial of Service (DoS) attacks that compromise system availability and disrupt critical services. Traditional intrusion detection systems often fall short in recognizing novel or evolving threats due to their reliance on static signatures and limited adaptability. This study proposes an enhanced hybrid machine learning model that integrates a supervised Random Forest (RF) classifier with an unsupervised Isolation Forest (IF) anomaly detector to improve detection accuracy and generalizability in IoT environments. Using a synthetic dataset, the model was evaluated across multiple performance metrics. Results indicate that the hybrid model outperforms standalone approaches, achieving 97.8% accuracy, 97.7% F1-score, and an AUC-ROC of 0.992. The hybrid architecture effectively balances the strengths of pattern-based classification and anomaly detection, reducing false positives while maintaining high detection rates. Additionally, the model demonstrates computational efficiency suitable for edge-based IoT deployments. These findings highlight the potential of hybrid learning frameworks to enhance the resilience and scalability of intrusion detection systems in resource-constrained IoT networks.*

Key Words: *Internet of Things (IoT), Denial of Service (DoS), Intrusion Detection System (IDS), Random Forest, Isolation Forest, Hybrid Machine Learning, Anomaly Detection.*

I.INTRODUCTION

The Internet of Things (IoT) continues to transform modern infrastructure by enabling seamless communication between devices across physical and virtual domains. According to recent industry estimates, over 30 billion IoT devices are expected to be deployed by 2025, intensifying concerns about data security and system resilience (Sharma et al., 2023). One of the most disruptive cybersecurity threats targeting IoT environments is the Denial of Service (DoS) attack, which compromises system availability by overwhelming target nodes with malicious traffic.

IoT devices often operate with limited processing power, low memory capacity, and lightweight protocols, characteristics that make them particularly susceptible to resource exhaustion attacks (Chen et al., 2021). In smart city networks, industrial control systems, and healthcare IoT, successful DoS campaigns can lead to economic losses, privacy breaches, and life-threatening consequences. Despite the implementation of firewalls and rule-based intrusion detection systems (IDS), many existing solutions fail to detect novel or evolving threats in real-time (Ali & Rehman, 2022).

Machine learning (ML) offers dynamic capabilities for pattern recognition and anomaly detection in network traffic. Supervised learning models like Random Forest have shown high performance in identifying known threats, but their dependency on labeled data limits their generalizability (Kumar et al., 2023). Conversely, unsupervised models like Isolation Forest detect anomalies without prior labeling but tend to generate false positives in noisy data (Bai et al., 2022).

To address these challenges, this paper presents an enhanced hybrid model that integrates Random Forest with Isolation